ANO V — Nº 53 — FEVEREIRO 1986 — Cr$ 14.000

Micro Sistemas

A PRIMEIRA REVISTA BRASILEIRA DE MICROCOMPUTADORES

OS COMPILADORES DA LINGUAGEM FORTH

O PADRÃO MSX NO BRASIL E NO MUNDO

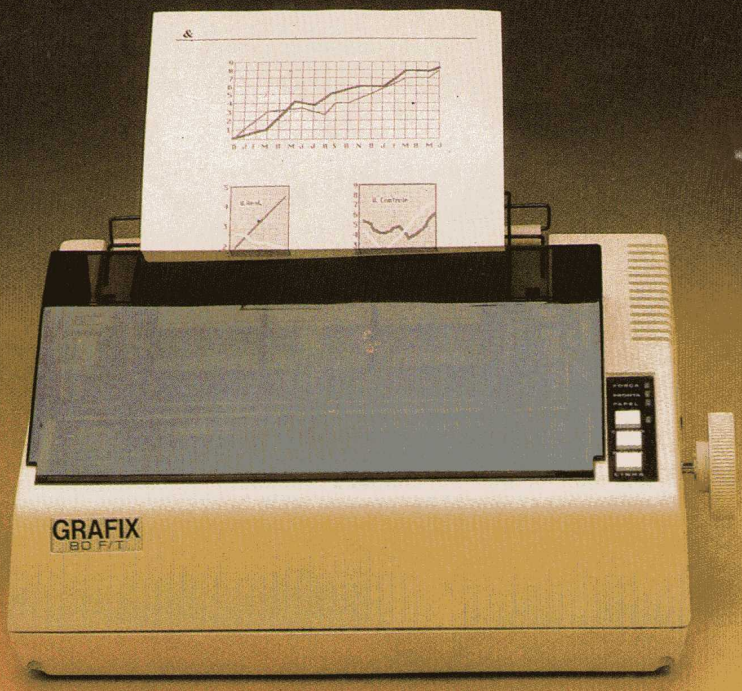
ISSN 0101 — 3041

CÓPIA DE PROGRAMAS

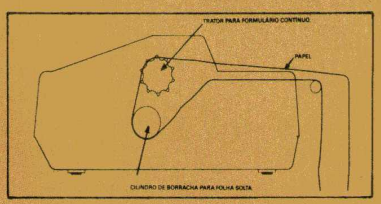
As estratégias de ataque e os truques da defesa

GRAFIX

GRAFIX 80 F/T. COMPATÍVEL ATÉ COM SEU PAPEL DE CARTA.



A GRAFIX 80 F/T é versátil. Com ela você pode facilmente imprimir formulários contínuos e folhas soltas como papéis de carta, envelopes, etiquetas, etc. Isto porque a GRAFIX 80 F/T é a única que vem equipada com trator e com cilindro de borracha, igual aos das máquinas de escrever, permitindo a impressão da primeira à última linha.



A GRAFIX 80 F/T é rápida e silenciosa. Imprime em 80 colunas a 160 caracteres por segundo com baixo nível de ruído.

Além de tudo isso, a GRAFIX 80 F/T é compatível com todos os microcomputadores, especialmente com os compatíveis IBM-PC, imprimindo todos os softwares disponíveis (Dbase III, Lotus 1-2-3, Word, ABC, Chartstar etc.)

Conheça a GRAFIX 80 F/T. A impressora certa para seu micro pessoal ou profissional.

SCRITTA

Al. Amazonas, 832 - Alphaville
Barueri - SP Tel. 421-1247

GRAFIX. TECNOLOGIA SEMPRE EM EVOLUÇÃO.

Micro Sistemas

Editorial

Abrir e proteger programas são temas constantes nas revistas especializadas de microcomputação, gerando muito interesse e retorno por parte dos leitores. A revista MICRO SISTEMAS, inclusive, teve um papel importante no mercado brasileiro por ter sido o primeiro veículo especializado a tratar diretamente do assunto, sem meias-palavras, publicando artigos e dicas sobre como proteger ou copiar software.

Em nossos primeiros anos, quando começamos a enfocar esta questão, balançamos o velho tabu do "impublicável": chegamos a receber cartas em que os leitores e colaboradores diziam "agora que vocês começaram a falar, eu também vou abrir o jogo".

Abrir o jogo? Sim, porque naqueles tempos o conhecimento de certos macetes era privilégio apenas de um pequeno círculo de usuários, e a pirataria era atividade bastante elitizada. Muita gente que ganhava dinheiro com a "indústria da cópia" não gostou nem um pouco quando começamos a desvendar os segredos do tesouro...

O fato é que através de um tratamento apropriado, tais assuntos foram sendo discutidos nas páginas de MS, e nosso leitor foi tornando-se consciente das vantagens e responsabilidades que o domínio daquelas técnicas lhe traria. Hoje, MICRO SISTEMAS apresenta para você a primeira parte de um texto bem completo sobre a defesa e o ataque na cópia de programas. O artigo vem assinado por Jean Pierre Lavelle, um profissional bastante conhecido e respeitado na área de microinformática, por sua atuação na Embratel e notadamente na arquitetura da rede Ciranda. Vale a leitura.

E para os muitos leitores que nos escreveram pedindo dicas sobre compiladores da linguagem Forth, apresentamos nesta edição uma reportagem com esclarecimentos.

Alda Campos

ANO V — Nº 53 — FEVEREIRO 1986

SUMÁRIO

6 CÓPIA DE PROGRAMAS: ATAQUE — Primeira parte do artigo de Pierre Jean Lavelle.

10 FESTIVAL DE DICAS NO TK90X — Artigo com técnicas e dicas de programação.

14 COMPILADORES FORTH — Reportagem.

16 PROTEGENDO DIREITOS AUTORAIS — Artigo de Clóvis Ático Lima Filho, para a linha Apple.

22 ARRANJO NO MICRO — Programa de Marcelo Contart, para a linha ZX81.

24 SISTEMA ASSEMBLER — Programa de Helvécio Carneiro Ribeiro Netto, para a linha TRS-80.

28 A INSTRUÇÃO CPIR NOS JOGOS — Artigo de José Armando Leite Fernandes, para a linha ZX81.

30 LÉO ATACA OS PIRATAS — Artigo de Léo Luiz Ferraz Netto, para a linha TRS-80.

48 CURSO DE FORTH (V) — Série de autoria de Antonio Costa.

54 O PADRÃO MSX — Primeira parte do artigo de Oscar Júlio Burd e Luiz Sérgio Y. Moreira.

BANCO DE SOFTWARE

32 ESTRATÉGIA
34 ANALISADOR DE HEADER
36 COMETA HALLEY
38 COPYDIR
40 PROTECTOR
42 SOLITÁRIO

SEÇÕES

4 CARTAS **50 LIVROS**

44 INICIANTE

18 BITS **52 DICAS**

CAPA: Maurício Veneza

Micro Sistemas

EDITORA / DIRETORA RESPONSÁVEL:
Alda Surerus Campos

ASSESSORIA TÉCNICA:
Roberto Quito de Sant'Anna; José Eduardo Neves; Luiz Antonio Pereira; Claudio José Costa.

CPD: Divino C. R. Leitão, Pedro Paulo Pinto Santos.

REDAÇÃO: Graça Santos (Subeditoria); Stella Lachtermacher; Mônica Alonso Monçores; Carlos Alberto Azevedo.

COLABORADORES: Aldo Naletto Jr.; Alvaro de Filippo; Amaury Moraes Jr.; André Koch Zielasko; Antonio Costa Pereira; Ari Morato; Cláudio de Freitas Bittencourt; Evandro Mascarenhas de Oliveira; Geraldo Simonetti Bello; Heber Jorge da Silva; Ivan Camilo da Cruz; João Antônio Zuffo; João Henrique Volpini Mattos; José Carlos Niza; José Ribeiro Pena Neto; Lávio Pareschi; Luciano Nilo de Andrade; Luis Carlos Eiras; Luiz Carlos Nardy; Marcel Gameleira de Albuquerque; Marcel Tarrisse da Fontoura; Maurício Costa Reis; Marcelo Renato Rodrigues; Mário José Bittencourt; Nelson Hisashi Tamura; Nelson N. S. Santos; Newton D. Braga Jr.; Paulo Sérgio Gonçalves; Rizieri Maglio; Rudolph Horner Jr.; Sérgio Veludo.

ARTE: Cláudia Duarte (coordenação); Leonardo Santos (diagramação); Maria Christina Coelho Marques (revisão); Wellington Silvares e Marco Antônio Rubim (arte-final).

ACOMPANHAMENTO GRÁFICO: Fábio da Silva

ADMINISTRAÇÃO: Janete Sarno

PUBLICIDADE

São Paulo:
Geni dos Santos Roberto
Contatos: Paulo Gomide;
Tels: (011) 853-3229, 853-3152

Rio de Janeiro:
Elizabeth Lopes dos Santos
Contatos: Regina Gimenez; Georgina de Oliveira
Tel: (021) 262-6306

Minas Gerais:
Sidney Domingos da Silva
Rua da Bahia, 1148 – sala 1318
CEP 30 000 – Belo Horizonte

Porto Alegre:
AURORA-ASSESSORIA EMPRESARIAL LTDA.
Rua dos Andradas, 1155 - Grupo 1606 - 16º andar
Tel.: (0512) 26-0839

CIRCULAÇÃO E ASSINATURAS:
Ademar Belon Zochio (RJ)
José Antônio Alarcon (SP) - Tel.: (011) 853-3800

COMPOSIÇÃO:
Studio Alfa, Coopim.

FOTOLITO:
Organizações Beni e Juracy Freire

IMPRESSÃO:
JB Indústrias Gráficas

DISTRIBUIÇÃO:
Fernando Chinaglia Distribuidora Ltda.
Tel: (021) 268-9112

ASSINATURAS:
No país: 1 ano: Cr$ 140.000

NÚMEROS ATRASADOS
Cada exemplar: Cr$ 8.000

Os artigos assinados são de responsabilidade única e exclusiva dos autores. Todos os direitos de reprodução do conteúdo da revista estão reservados e qualquer reprodução, com finalidade comercial ou não, só poderá ser feita mediante autorização prévia. Transcrições parciais de trechos para comentários ou referências podem ser feitas, desde que sejam mencionados os dados bibliográficos de MICRO SISTEMAS. A revista não aceita material publicitário que possa ser confundido com matéria redacional.

FILIADA AO

IVC

MICRO SISTEMAS é uma publicação mensal da

ATI Análise, Teleprocessamento e Informática Editora Ltda.

Endereços:

Rua Oliveira Dias, 153 - Jardim Paulista - São Paulo/SP - CEP 01433 - Tels: (011) 853-3800 e 881-5668 (redação)

Av. Presidente Wilson, 165 - grupo 1210 - Centro - Rio de Janeiro/RJ - Tel: (021) 262-6306.

cartas

O sorteado deste mês, que receberá uma assinatura anual da revista MICRO SISTEMAS, é Leo Lopes Leite, de Maringá – PR.

SEM RODEIOS

No número 50 de **MS**, encontro o "velho-jovem-combativo" Pierluigi Piazzi num artigo que, se nosso país ainda tiver fôlego para tornar-se sério, conservará como documento histórico. Resumindo: Pierluigi condensa a discussão de uma problemática aguda – brasileira, profundamente brasileira, porém terceiro-mundista por todos os aspectos! – em apenas uma página e meia. Delata, com todas as palavras, com descortino, sinceridade e – principalmente – coragem, as falsetas, escorregões, salafrarices, incompetências, enganos e covardias de nossos industriais da Informática (...)

Pierluigi foi além da mera denúncia; MICRO SISTEMAS arriscou ainda mais! Cutucou os tigres com vara muito curta, revelando estar – ao contrário das demais publicações da área, somente preocupadas com aspectos econômicos-financeiros das empresas ligadas à Informática – ao lado dos usuários e criadores da máquina e do soft (...)

Com os respeitos de seu velho admirador,
Nilson D. Martello
Rio de Janeiro – RJ

Apreciamos bastante o artigo onde o Sr. Piazzi faz uma análise objetiva do mercado brasileiro, em linguagem sem rodeios, direta ao alvo. No entanto, (toda carta tem um no entanto), existem dois pecadilhos (não chegam a ser pecados), que deveriam chegar ao conhecimento do público em geral: 1) a maior biblioteca de software disponível é a do TRS-80-III, e não a do Apple; 2) o Apple não faz uso do Index Hole do disquete, tornando-se assim, muito mais vulnerável a erros de disco.
Paulo A. Guarinello
Micromaster Ltda.

BAIXANDO A RAMTOP

Estou desenvolvendo um programa BASIC em um CP-200 que necessita da RAMTOP no endereço 31815. Infelizmente, só consigo baixá-la através de POKE 16388, 71 POKE 16389, 124 NEW, para então carregar o programa com LOAD "". Caso contrário, os últimos bytes da RAMTOP são invadidos pelo programa.

Fiquei entusiasmado ao ver a dica *Mude o topo da RAM*, publicada em **MS** nº 45, mas observei um inconveniente: o programa é necessariamente listado após sua execução. Assim, gostaria de uma rotina que fosse colocada dentro de um programa e que, após o abaixamento da RAMTOP, o mesmo continuasse sua execução normalmente.

Correspondências para a rua São Paulo, 1225, Centro, CEP 60000.
José Carlos Elcias
Fortaleza – CE

MS AGRADECE

Primeiramente gostaria de parabenizá-los pelo excelente nível de **MICRO SISTEMAS**. Achei muito interessante o artigo de Antonio Eduardo Costa Pereira, publicado no número de agosto, sobre "As Linguagens do Futuro".

Gostaria de trocar idéias com usuários do CP400 Color. Correspondências para a Av. Princesa Leopoldina, 298, apt. 1101, CEP 40.000, Graça.
Norma Sueli F. de Araújo
Salvador – BA

CORRESPONDÊNCIAS

Gostaria de trocar correspondências com leitores que possuam micros da linha Apple, principalmente o TK2000. As cartas devem ser enviadas para o seguinte endereço: H 27D, nº 108, CTA.
José B. Silveira Jr. (S. J. Campos – SP)

Possuo um MC 1000 e gostaria de me corresponder com pessoas que possuam o mesmo equipamento. Meu endereço é: Rua Martim Afonso, 1335, CEP 87100.
Leo Lopes Leite
Maringa - PR

PONTO DE VISTA

É de conhecimento de todos que a automação bancária contribui para agilizar os serviços bancários, beneficiando os clientes através dos bancos eletrônicos e caixas automáticas. Os correntistas têm plena garantia dos seus cheques. Se forem extraviados ou roubados, basta notificar o banco para assegurar seu patrimônio. Agora pergunto: que segurança tem atualmente o pequeno (ou mesmo o grande) comerciante que recebe um cheque roubado? Será que já não é hora de se dedicar algum esforço para resolver este problema?

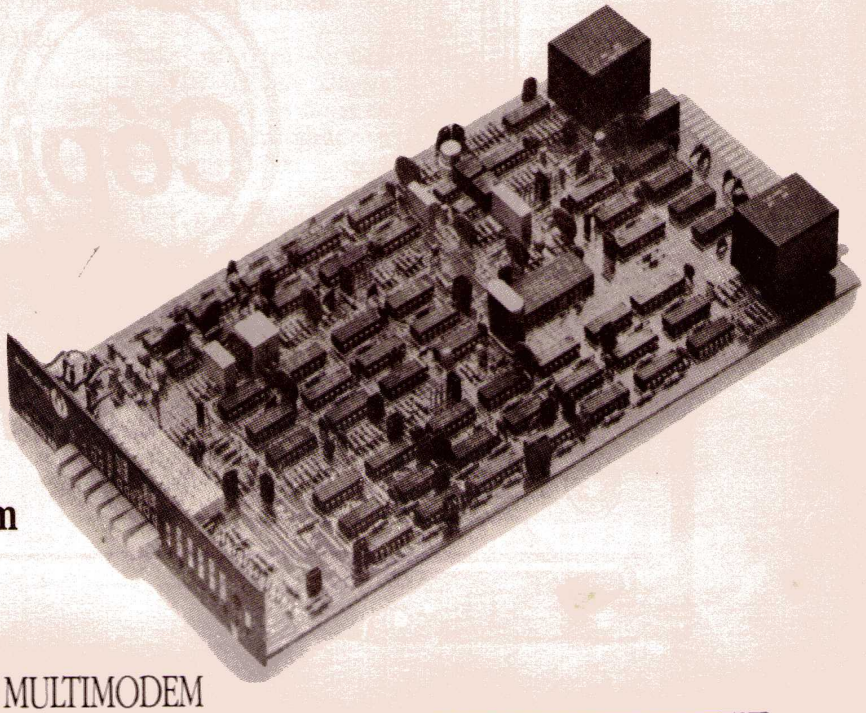
Hoje, pedir o cartão de garantia, RG ou o telefone é totalmente inútil. Tentar telefonar para a agência bancária é impraticável pelo tempo que se gasta até obter uma resposta; os documentos em muitos casos são falsificados; os cartões magnéticos (ou de garantia) não identificam a pessoa física; os cheques não apresentam o número de telefone da agência para consultas. Os terminais de compras poderiam resolver o problema, mas há dúvidas quanto ao custo da instalação. E ainda há mais questões: e as padronizações? Será que precisaremos instalar tantos terminais quanto forem os sistemas?

Disso tudo, concluo que do ponto de vista daquele que recebe um pagamento em cheque, a automação bancária de nada mais serve, além de ser chamariz para atrair mais e mais correntistas através de pomposos anúncios na TV.
Ernesto Seguchi
Rio de Janeiro – RJ

Envie sua correspondência para: ATI - Análise, Teleprocessamento e Informática Editora Ltda., Av. Presidente Wilson, 165/gr. 1210, Centro, Rio de Janeiro/RJ, CEP 20030, Seção Cartas/Redação MICRO SISTEMAS.

LANÇAMENTO

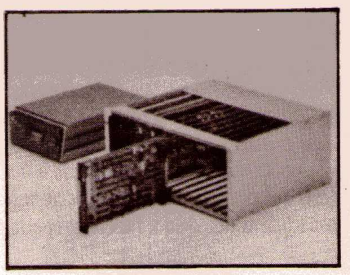
Liberdade de Escolha



EM-1275
Multimodem - O modem profissional também para micro.

A ELEBRA está lançando o MULTIMODEM - o modem que vale por três. Um equipamento profissional que você também poderá usar com o seu micro.

Ele opera em 300 bps, 1200 bps e 1200 bps com canal secundário de 75 bps, para que você tenha a liberdade de interligar-se a qualquer rede de comunicação de dados ou qualquer Tele-serviço.



- Resposta automática (opcional)
- Opera em linhas comutadas ou dedicadas a 2 ou 4 fios
- Versão mesa ou bastidor
- Desconexão automática ao término da transmissão
- Loops (opcional)
- Padrão de teste
- Chave voz/dados
- Segue as recomendações CCITT e TELEBRÁS.

EM-1275 MULTIMODEM. A alta tecnologia em modem, multiplicada por três. Para você ter liberdade total de escolher a melhor maneira de se comunicar.

Modems ELEBRA. Produtos com passado, presente e muito futuro.

elebra telecon

Empresa controlada por Docas S.A.

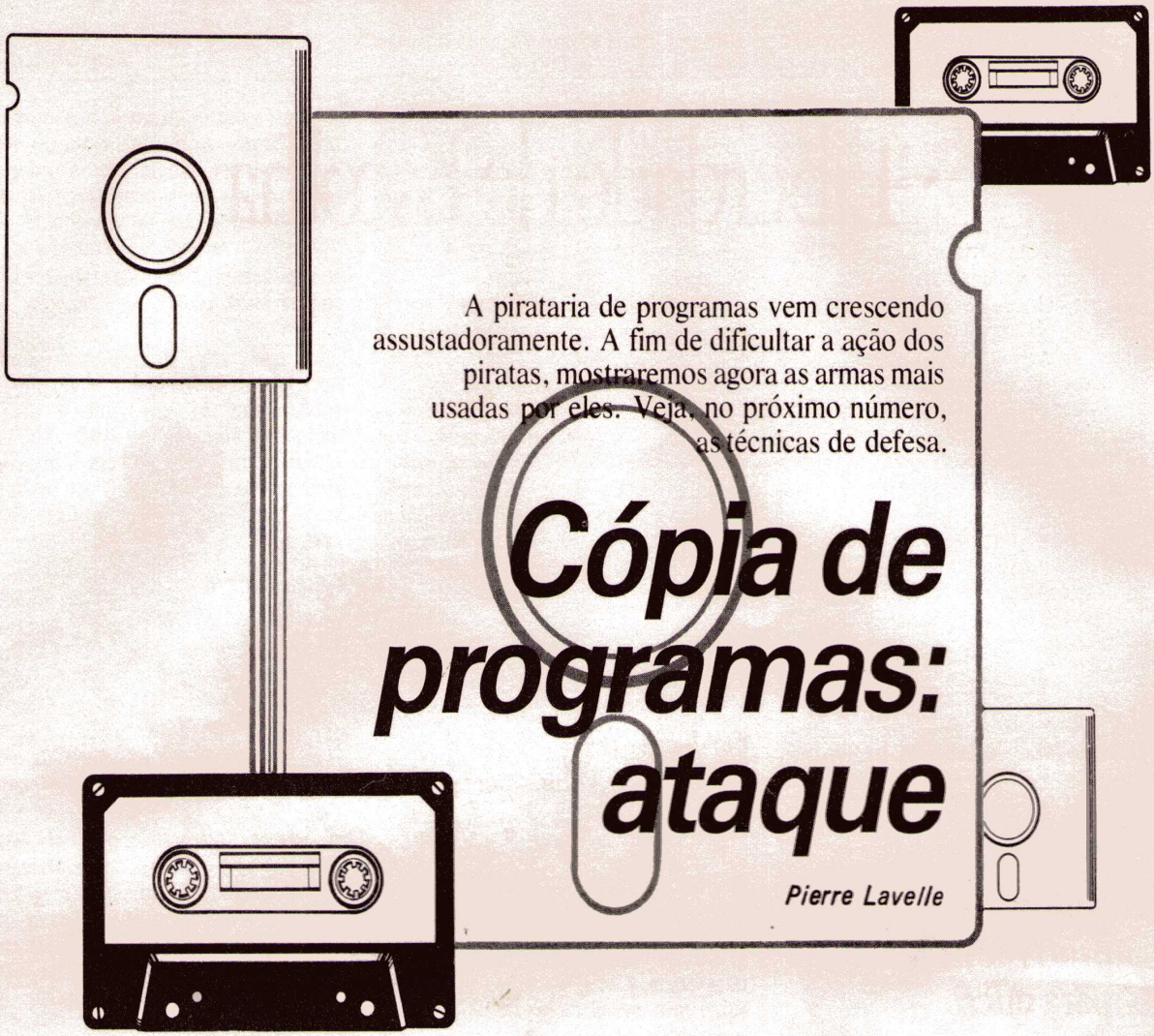
Vendas: Av. Eng.º Luiz Carlos Berrini, 1461,
São Paulo - SP CEP 04571 - Fone (011) 533-9977
Telex (011) 25131
Filial Rio: Av. Rio Branco, 50 - 11.º andar, CEP 20090
Fones (021) 233-0223/233-2220/233-3977.

Filiada à Abicomp

À ELEBRA S.A. - AV. RIO BRANCO, 50 - 11.º AND. - CEP 20090 - RIO DE JANEIRO - RJ — EM

☐ Solicito o envio de folheto
☐ Solicito visita de um Representante

Nome .....
Empresa ..... Cargo .....
Endereço .....
Telefone ..... Ramal ..... Cidade ..... Estado .....



A pirataria de programas vem crescendo assustadoramente. A fim de dificultar a ação dos piratas, mostraremos agora as armas mais usadas por eles. Veja, no próximo número, as técnicas de defesa.

Cópia de programas: ataque

Pierre Lavelle

Desde os tempos mais remotos da História da humanidade, a criação e acumulação de riquezas geram cobiça naqueles que não as possuem e, conseqüentemente, o medo de perda naqueles que as têm. Os regimes políticos se sucedem, mas a noção de Direito de Propriedade permanece, baseada em leis que a fazem respeitada.

A chegada de tecnologias, permitindo a duplicação fácil e barata de certos tipos de riqueza, recoloca em discussão os fundamentos do direito de propriedade: se roubam meu anel de ouro, perco-o para sempre; mas se duplicam meu livro, disco, filme em cassete e me deixam o original, o que é que eu perco? Sendo assim, o que tenho a perder em facilitar a duplicação? Certamente ganharei amigos!

O computador veio exacerbar a discussão: uma vez que a tecnologia digital empregada reconstrói o sinal a cada cópia (o que não acontece com livros, discos ou filmes), não é mais possível diferenciá-la do original. E, por incrível que pareça, às vezes, a cópia é de melhor qualidade! A divulgação do computador colocou na mão do público um poder de duplicação sem precedente na História, como se todo mundo tivesse uma impressora de cédulas de dinheiro em casa. Quantas cédulas falsas estariam em circulação?

Em termos práticos, as pessoas pensam: "Por que pagar se posso copiar?". Quantos advogados convictos do direito de propriedade intelectual têm gravador e/ou videocassete em casa, sem falar das pilhas de cópias xerox? As respostas não são nada evidentes: na novela jurídica da duplicação de discos e de fitas de videocassete a última palavra dos tribunais americanos é que, se for para uso pessoal, não é criminoso (não me arrisco a comentar a jurisprudência brasileira a esse respeito). Existe uma diferença legal entre cópia eletrônica analógica e cópia eletrônica digital? Existe mesmo uma clara diferença técnica? E quando, num futuro próximo, a Engenharia Genética começar a realizar suas mais recentes promessas de *clonação*, assistiremos boas discussões!

Neste artigo, não pretendo tratar de todas as maneiras moralmente duvidosas de se usar um microcomputador, mas apenas da duplicação de programas. O assunto é vasto o bastante, e a tecnologia evolui rapidamente.

A minha contribuição será a de demonstrar quais são os métodos usados para atacar e quais são as defesas. O conhecimento dessas últimas é muito pouco difundido e espero que a leitura desses artigos aumente seu nível tecnológico: é triste tentar defender-se com escudos de papel quando estão atacando com verdadeiras bazucas.

O ATAQUE

Como toda profissão, a *pirataria* também tem suas ferramentas especializadas. Vamos passar o arsenal em revista para analisar as possíveis defesas.

O backup e as senhas – Todo sistema tem seu modo de copiar arquivos ou discos inteiros: procedimento altamente recomendável quando os programas ou dados têm qualquer importância. Quase todos têm uma forma de verificar se os arquivos a copiar são legalmente copiáveis, mas os meios de driblar esses mecanismos de proteção já foram amplamente divulgados pelas revistas.

Contra-ataque: Não há. São apenas mecanismos para ajudar o usuário a se defender dos seus próprios erros.

O clone – Com nomes pitorescos como *Copycat, Capyiipc, Copywrit* (jogo de

palavras com *Copyright*), *Nibbles Away* (jogo de palavras com a música *Anchors Away*), existem vários programas especializados em copiar fisicamente discos inteiros, na base da leitura de uma trilha inteira, entendimento do seu modo de formatação e recriação em outro disco. Geralmente eles não são copiáveis, a menos que caiam vítimas de um melhor ainda, o que é apenas questão de tempo. O argumento dos vendedores é que os utilitários de cópia dos fabricantes são lentos demais, o que é verdade. Todos previnem na capa que o uso para copiar programas comerciais é vedado pela lei... norte-americana. Nenhum deles existe de forma oficial no país, porém são mais difundidos do que programas de banco de dados.

Contra-ataque: Usar um tipo de formatação de trilhas desconhecido deles. Infelizmente, este recurso hoje é ilusório. Tem de se recorrer a técnicas de controle da disseminação de cópias, tratadas mais adiante.

O ZAP – É um programa que permite examinar e modificar qualquer byte de qualquer setor de qualquer disco formatado normalmente. Todo sistema tem o seu, com nomes bem parecidos: SuperZap, DiskZap, AsciiZap, ou então derivados como Debug (literalmente: tira defeitos, já que, em inglês, o *Bug* ou inseto é o apelido do erro de programa) ou DDT (esse é um jogo de palavras: o DDT, cientificamente chamado de Dicloro-Dimetil-Tetraeno, serve para matar os *bugs*...)

Este programa permite trazer para a memória um setor qualquer do disco, por seu endereço absoluto ou relativo dentro de um arquivo; visualizar o conteúdo em hexadecimal e em caracteres normais; aplicar fórmulas sob todos os bytes (tal como, inverter os bits, somar um valor, executar um OR Exclusivo com um valor) e, posicionando o cursor sobre um dado, modificá-lo à vontade e gravar o setor atualizado de volta no disco.

Cada um destes programas tem suas qualidades: o pirata consciencioso chega a fazer uma versão pessoal que admite formatações de disco tão complicadas quanto os mecanismos de proteção encontrados.

Numa máquina com vários usuários simultâneos, estes programas permitem observar e até agir sobre as áreas de memória ditas protegidas. Todos os fabricantes tem um programa parecido, normalmente de uso apenas interno, somente para a depuração do sistema. Mas há vazamentos. De posse da chave correta ou aproveitando um furo da arquitetura, o pirata poderá observar o seu bel-prazer a execução do programa visado.

"Além de linguagens de programação, é fundamental conhecer bem a arquitetura do equipamento."

Contra-ataque: Usar estruturas de dados propositalmente complexas, como, por exemplo, um código próprio para representar mensagens que não seja o ASCII. Sugestão: codificar a N-ésima letra de uma mensagem como rotação ((valor ASCII + constante + N) XOR endereço) ou qualquer outra fórmula invertível.

Conhecimento da linguagem de máquina – Além de ser versado em algumas linguagens de programação, é fundamental ter bons conhecimentos da arquitetura do equipamento a ponto de se ler diretamente a linguagem interna na sua forma hexadecimal e de ter em mente a ação de cada instrução sobre os *flags* da Unidade Central. O caminho normal é tentar entender o código gerado pelos compiladores disponíveis, para depois escrever seus próprios programas em linguagem de máquina.

Contra-ataque: Não ensinar a linguagem montadora. Aos usuários basta conhecer o modo de uso dos pacotes essenciais: ... calc, word..., ...base, ...term e ...graph. O passo seguinte é aprender uma linguagem de alto nível: Pascal, C, BASIC etc. O ensino do Assembler deveria ser restrito aos programadores experimentados, que, por terem sofrido bastante, serão menos propensos a abusar do poder da ferramenta do que os jovens amadores geniais. As faculdades de Medicina têm experiência multissecular em filtrar os candidatos a cirurgião. Nossa profissão tem muito o que aprender sobre isto.

Conhecimento de entradas e saídas – Além da familiaridade com o jogo normal de instruções, é fundamental entender perfeitamente como se processam as entradas e saídas e as interrupções. Os manuais são raramente explícitos a respeito (o autor tem razões para desconfiar que seja proposital: ele também agiria desta maneira). Muito pode ser aprendido decompilando-se um sistema operacional (a começar pelo *Bootstrap*, ou carregador, que dá a partida à máquina) e muito mais decompilando-se os programas de autoteste do computador.

Contra-ataque: Não difundir esta informação. Tornar os folhetos que descrevem o modo de uso das pastilhas de entrada e saída raridades, ao invés de incluí-las no manual. Usar pastilhas exóticas ligadas ao equipamento por modos de endereçamento e de interrupção quase loucos. Alguns fabricantes conseguiram isto, talvez sem querer.

Freeway data buffer

APROVADO PELA SEI
Portaria 205 de 23-08-85

PSI informática

Avenida Pavão, 346 - Indianópolis - CEP 04516 - São Paulo - SP
Telefone: (011) 533-0120 / 533-0112 – Telex: 22966

Aumente a produtividade do seu micro, acabando com a espera pela impressora.

FREEWAY é um "data buffer" que controla sozinho a impressora durante a emissão dos seus relatórios, liberando o micro para outras tarefas.

- Memória de 64 Kb expansível a 576 Kb.
- Opera em modo comprimido, permitindo armazenar relatórios muito maiores que sua memória.
- 9 funções: CÓPIAS ADICIONAIS SEM USO DO MICRO, AVANÇO DE PÁGINA, PAUSA AUTOMÁTICA POR PÁGINA, ESTATÍSTICAS, MODO DIRETO, AUTOTESTE, etc.
- Comunicação serial (RS 232), paralela (Centronics) e combinações, permitindo ligar micro serial em impressora paralela e vice-versa.
- Pode ser ligado em terminais, plotters, RJE, etc.

Os decompiladores – As fontes de muitos programas não são encontráveis por duas razões: por motivo de proteção, para tentar impedir a divulgação dos conhecimentos embutidos no programa ou por simples vergonha, caso o programa seja tecnicamente um horror.

Como o conhecimento de muitos programas é em si instrutivo, por mostrar técnicas de programação diferentes, existem de longa data programas decompiladores, cuja função é produzir em papel ou em disco uma listagem inteligível de um programa em binário, residente na memória ou em disco.

a) Os compiladores básicos tentam dar o sentido de instrução a cada byte do arquivo de entrada e produzem uma listagem obviamente errada, quando encontram uma área de dados ou perdem a sincronização no caso de instruções de comprimento variável.

Contra-ataque: Não usar programas em alto nível, como BASIC compilado, pois o código gerado tem muita regularidade e poucas sutilezas, sendo quase tão fácil de ler quanto o original, uma vez empregado o mesmo compilador sobre alguns programas de teste e observado o resultado.

b) Os decompiladores avançados são avisados pelo usuário dos vários pontos de entrada do programa e seguem as pistas assim iniciadas. A cada bifurcação (pulo condicional, chamada de sub-rotina) eles marcam numa pilha uma das duas direções e seguem a outra até encontrar um terreno já varrido. Então, seguem a trilha marcada pelo topo da pilha até a pilha ficar vazia. Os dados que sobraram são mesmo interpretados como dados e não como instruções. O principal defeito é quando encontram um pulo calculado (JP (HL)). Neste caso, normalmente, eles não seguem a pista, mas deixam na listagem uma advertência: o usuário deverá usar o conhecimento das instruções precedentes a este pulo para concluir quais são os valores razoáveis, na hora da execução do programa examinado, e pedir uma nova decompilação com umas pistas iniciais a mais.

Contra-ataque: Usar e abusar dos recursos do Assembler: fazer o programa se automodificar; inicializar a pilha com endereços e dados e dar RET e POP, ao invés de JP e LD; calcular, de modo complicado, endereços de rotinas e, enfim, usar as instruções não documentadas do micro (o Z80 tem quase o dobro das instruções constantes dos manuais).

Os interpretadores – Estes programas simulam a execução de um outro e são de grande valia na depuração. Entretanto, eles podem ser usados para outros fins: a) os clássicos interpretam o programa, instrução por instrução, incluindo o funcionamento de todos os registros (no caso do Z80, incluir os registros I e R), mostrando na tela várias janelas sobre os registros, instruções, pilhas e dados, e podem parar quando encontram certos valores em alguns registros ou memórias, ou quando encontram certas instruções de entrada e saída. O problema é que estas entradas e saídas não podem ser realizadas em tempo real, devido à interpretação. Para simular convenientemente uma máquina de certo porte, numa máquina igual, é necessário usar técnicas de máquinas virtuais e paginação do espaço de memória simulado em disco, ou melhor, dispor de uma máquina de porte maior, como, por exemplo (o que seria o cúmulo da pirataria), usar uma cópia (ilegal) de um micro de 16 bits para simular o funcionamento de uma cópia (ilegal) de um certo programa numa cópia (ilegal) de um micro de 8 bits, provavelmente a fim de encontrar e derrubar o mecanismo de proteção!

"O conhecimento de muitos programas é instrutivo por mostrar técnicas de programação variadas."

Contra-ataque I: Apostar na imperfeição do interpretador especialmente em operações envolvendo o tempo real: se é fácil simular o comportamento de um registro **R** (de Refresh), no Z80 é muito mais difícil simular as entradas e saídas, especialmente em disco, em tempo real. O programa pode depender do registro **R**, não ter dado um número de voltas inferior a **X** ou superior a **Y** entre voltas do disco, usando o valor num pulo calculado.

Contra-ataque II: Usar um programa inicial que carregue uma massa de dados, aplicar sobre ela um certo algoritmo (XOR, rotações...) ao mesmo tempo em que verificar se o disco continua girando e, assim, após ter reconstruído o segundo programa, dar-lhe o controle de modo disfarçado: a primeira instrução do segundo programa sendo montada após a última do primeiro, por exemplo. Este por sua parte vai repetir com um terceiro, e assim por diante, até chegar no verdadeiro programa.

b) Os emuladores em linha, placas auxiliares que permitem o controle passo a passo do microprocessador, dão ao usuário a facilidade de mexer nos registros e na memória sem diminuir de um só byte o tamanho da memória nem alterar em um só microssegundo o tempo de execução. Essas placas são feitas de eletrônica discreta que emulam o comportamento do micro, mas com todos os controles desejáveis, na outra extremidade de um cabo que se ligue no lugar da Unidade Central do micro. A grande vanta-

O pirata

Por que este apelido? A origem é confusa, afinal nunca ouvi falar de alguém usando força bruta para apropriar-se da cópia de um programa. Por outro lado, o dono original nunca perde a posse de um bem enquanto, no tempo dos piratas verdadeiros, as vítimas perdiam algo (freqüentemente a vida). Hoje, o pirata não usa de força nem sua vida perde algo tangível.

Admitindo – com restrições – este vocábulo, vamos passar a categoriazá-los, por ordem crescente de sofisticação:

O pirata ocasional – *Somos todos nós que, lendo um artigo interessante, tiramos uma "cópia xerox", ao invés de comprar outro exemplar; neste gesto corriqueiro, negamos as noções de Direito Autoral, Respeito à Propriedade Intelectual etc.*

O prolongamento natural é, vendo rodar um programa interessante, puxar um disquete virgem e pedir inocentemente uma cópia. Os mais honestos cidadãos não hesitam nisso, daí a pergunta: se esta conduta não perturba a enorme maioria da população, por que a chamar de imoral? Antes dos juristas responderem apressadamente, deveríamos ouvir os filósofos...

O pirata amador – *Após ter encontrado alguma dificuldade em duplicar certos programas, o pirata ocasional começa a prestar mais atenção às colunas de dicas dos numerosos jornais especializados, à venda em todas as esquinas; passa a ler mais a fundo os manuais técnicos; enriquece sua biblioteca com vários programas utilitários e começa a acumular conhecimentos especializados. Cada nova conquista cria a vontade de ir mais além (como dizia Hillary, voltando do Everest: "Escalamos as montanhas apenas porque elas estão aí").*

O grande pirata amador – *Chegando ao cume, nosso amador conquista uma nova visão do mundo, passando a buscar a dificuldade em si, vai atrás dos programas difíceis, com o objetivo exclusivo de furar as medidas de proteção. Falar para ele que tal programa é incopiável é o mesmo que sacudir um pano vermelho na frente de um touro: provocado, ele vai passar noites a fio inclinado sobre o problema. Feita a conquista, ele provavelmente jogará o programa fora, já que não tem normalmente nenhum atrativo para ele, após dar algumas cópias de presente a seu círculo de amizade, o qual às vezes inclui uma rede de computadores. No fundo, a consciência dele não é pesada: os culpados, se culpados há, são aqueles que usam as cópias ilegais.*

O pirata profissional – *Premiados pelas circunstâncias financeiras ou simplesmente percebendo que existe um mercado – e absolutamente nenhum tipo de risco – nosso amador adquire, ou fabrica ele mesmo, alguns equipamentos especializados e coloca seus serviços à disposição mediante retribuição. Assumindo sua condição de* gatilho de aluguel, *chega a dar consultoria sobre como se proteger dos piratas, escreve livros etc. Este, sim, causa dano econômico ao criador do programa; mas em qual lei enquadrar o crime? Como provar em juízo que esse programa, conversando em bom português, não passa de uma tradução retocada desse outro programa que fala inglês?*

gem é a execução em tempo real das entradas e saídas, sem perda de controle da execução do programa, permitindo o armazenamento da imagem de memória e dos registros após a execução dos passos de proteção. Tal imagem pode facilmente ser convertida num arquivo copiável e perfeitamente executável.

Contra-ataque: No plano técnico, infelizmente não se conhece nenhum. Tais placas não se encontram nas lojas, mas podem ser feitas por qualquer estudante de engenharia eletrônica.

O tradutor – A posse do binário executável e desprotegido é quase o bastante para os piratas amadores. Falta apenas a tradução para o português. Essa pode ser feita de modo trivial, com o ZAP, desde que as palavras novas tenham o mesmo comprimento das palavras originais. Isto explica a profusão de programas falando um português esquisito, onde se escreve *agora* no lugar de *pronto*. Como *ready* tem apenas cinco letras, *pronto* não cabe.

Contra-ataque: Usar letras no meio das frases como constantes ou instruções; calcular periodicamente somas de controle do próprio programa.

O grande pirata amador vai querer ir além. Ele vai decompilar o programa, reparar a cadeia de caracteres, trocar por outras de tamanho possivelmente diferente – se precisar e se puder (às vezes, a formatação das telas não deixa muita margem), e recompilar. Após fazer isso manualmente algumas vezes, ele vai construir um programa tradutor, para automaticamente localizar as cadeias de caracteres no código decompilado e fazer os ajustes necessários, uma vez escolhida a cadeia substituta.

Contra-ataque: Fazer o programa trabalhar com áreas de manobra cujos endereços são calculados por um algoritmo não evidente, de tal maneira que essas áreas não podem ser deslocadas nem alargadas. Entre elas, são colocadas as palavras de comprimento a ser protegido. E o algoritmo de soma de controle tem de ser muito bem disfarçado, por exemplo, junto com as instruções de varredura de tabelas.

O disfarçador – O pirata profissional tem mais um problema a resolver. Como os reais donos do programa vão provavelmente partir para a briga, ele tem de disfarçar o produto para poder se defender em juízo, caso o Meritíssimo Sr. Juiz seja mais exigente do que o do caso Sinclair.

Para isso, existem disfarçadores: programas que investigam um outro programa na forma de linguagem fonte (possivelmente produzida por um decompilador) e anotam os trechos executados seqüencialmente: eles começam por um rótulo e terminam por um pulo incondicional. Os pulos *curtos* são expandidos para pulos *longos*, e os trechos são armazenados separadamente.

Se for necessário, os pulos condicionais podem ter seu teste invertido, com o trecho seguinte ao do teste original, passando a compor um bloco separado, e o trecho alcançado sob condição passando a fazer parte do bloco atual.

Se existem N trechos assim, é possível recombiná-los de N! maneiras diferentes, sem alterar a lógica e o funcionamento do programa. N, tipicamente, é um número de dois ou três algarismos (o número fatorial 100! tem 158 algarismos e o número fatorial 1000! tem nada menos que 2.568.

"Como o dono do programa pode querer partir para a briga, o pirata tem de disfarçar o produto."

A recombinação pode ser feita ao acaso ou, então, procurando juntar um trecho que acaba com um desvio incondicional para um certo rótulo com um trecho que começa por esse rótulo, permitindo a retirada da instrução de desvio.

Após a recombinação, passos sucessivos (teoricamente dois) trocam os pulos *longos* para pulos *curtos*, quando for possível, para poupar memória. Se for o caso, o disfarçador pode manter idênticos os endereços iniciais de certas rotinas e/ou aplicar as técnicas de combinação de programas para eliminar trechos comuns no fim de blocos diferentes etc.

A beleza (?) do método é que um programa assim *mastigado* pode ficar menor, mais veloz e, a nível do código-objeto, completamente diferente do original, sem que o pirata tenha a menor necessidade de entender a lógica do programa, nem mesmo ver o fonte, se não foram aplicadas as contra-medidas cabíveis.

Contra-ataque: O programa realiza uma varrida dele mesmo, calculando somas de controle a partir do endereço e do gênero das instruções encontradas. Essa varrida pode se dar sub-repticiamente, passo a passo, a cada caráter digitado e/ou emitido para a tela, ou ainda junto com os acessos ao disco.

O modem com discagem automática – Filmes recentes divulgaram história sobre Menores, Micros, Modems e Militares. Na realidade, histórias parecidas acontecem todos os dias: deixou de ser notícia o fato de uma criança americana ser pega em flagrante usando seu micro caseiro para furar o bloqueio de um longínquo computador sem seus pais perceberem. Entretanto, ainda é notícia quando prendem de uma tacada só um clube inteiro (clube dos 800, em Minneapolis). O método empregado é o seguinte:

- primeiro, o micro é programado para discar todos os números de telefone das redondezas, procurando ouvir um tom característico de modem após, no máximo, dois toques. Ao ritmo de três testes por minuto, mais de 4000 telefones são testados por dia; mais de 100.000 por mês. A conta não será tão alta assim: poucas pessoas atendem ao segundo toque. (Todos nós já fomos acordados no meio da noite por um telefone que pára sozinho, após dois toques: seria esta a explicação?)
- devemos notar que esta primeira fase é toda acadêmica: a lista dos números quentes circula no colégio.
- a seguir, esses números são discados manualmente para se sentir que tipo de modem está do outro lado. Após essa filtragem manual, o micro volta a agir e tentar todas as combinações de senhas possíveis, enquanto seu dono dorme tranqüilamente. Ainda existem sistemas cujas senhas têm apenas quatro ou cinco letras, que podem ser descobertas em poucas horas.

Contra-ataque I: Após um máximo de três tentativas de apresentação da senha, o sistema deveria cortar a ligação, e, se ela voltasse a se repetir em poucos minutos, deveria soar um alarme ao operador. Um método bem mais seguro usa a discagem reversa: o gerenciador de comunicações do sistema central deve pedir a identificação e imediatamente pedir para desligar. Após isso, ele verifica no cadastro qual é o número de telefone associado a esta identificação e disca de volta, usando uma outra linha. Ele pede e verifica a senha, e só então a ligação com o sistema central é completada. É necessário que este gerenciador seja independente do sistema para não poder ser atacado por trás, por algum programa traidor.

Contra-ataque II: Manter nas alturas o custo dos equipamentos e proibir os BBS particulares e os modems com discagem automática.

■

Pierre Jean Lavelle é Engenheiro formado pela École Nationale Supérieure d'Electrinique, d'Informatique et d'Hidraulique de Toulouse e Doutor em Matemática (extensão em Informática) pela Université de Toulouse. Perito da Cooperação Técnica Francesa e Professor Adjunto da COPPE/UFRJ, Lavelle trabalha atualmente na Equipe de Redes Locais do Departamento de Processamento de Dados da EMBRATEL, tendo sido, inclusive, Arquiteto da Rede Ciranda.

Melhorar a impressão no vídeo, criar gráficos em alta resolução ou proteger programas são algumas dicas que este artigo lhe oferece.

Festival de dicas no TK90X

André Koch Zielasko
Alexandre Mariano de Brito

Nada melhor do que uma boa dica quando se está diante de um problema ou mesmo quando é necessário aprimorar uma idéia.

Para auxiliá-lo nesta tarefa, MS reuniu algumas sugestões enviadas por nossos colaboradores. Através delas, você conhecerá um pouco mais dos recursos que o TK90X lhe oferece e aprenderá ainda a explorá-los melhor.

Portanto, se você é um usuário do TK90X, seja *calouro* ou *cobra*, estas rotinas poderão ajudá-lo a colocar seus conhecimentos em prática, ou, no mínimo, distraí-lo.

Por isso, mãos à obra, ou melhor, ao micro!

1 – IMPRESSÃO

Uma característica particular do TK90X é o uso do apóstofo('): quando ele é utilizado, a próxima impressão tem início na linha seguinte (no começo). Podem ser usados tantos apóstrofos quantos desejados, sendo que cada um corresponderá ao pulo de uma linha. Para verificar isto, rode o programa abaixo e repare o resultado.

```
10 PRINT "uma linha"
20 PRINT "outra linha"
30 PRINT "uma linha"''"outra li
nha"
40 PRINT ''"pula 2 linhas"
```

É possível também imprimir nas linhas de edição, como mostra o programa a seguir:

```
10 FOR i=1 TO 5
20 PRINT #0;TAB i;"linha de ed
icao=";i
30 PRINT "linha de programa=";
i
40 NEXT i
50 PAUSE 0
```

Você deve ter notado que, ao final da execução, aparecem escritas apenas as mensagens da linha 30, enquanto que a da linha 20 é apagada. Acrescente a linha **50 PAUSE 0** ao programa e rode-o novamente. Observe que a instrução **PAUSE 0** interrompe o programa até que qualquer tecla seja pressionada, permitindo que a mensagem da linha de edição seja visualizada.

Outro recurso que pode ser utilizado para se escrever na linha de edição é dar POKEs na área de memória reservada para o vídeo. Rode e analise o programa a seguir (os endereços podem ser encontrados no manual).

```
10 LET a$="uma linha"
20 FOR n=1 TO LEN a$
30 LET b=CODE a$(n)
40 LET c=15360+b*8
50 LET d=20671+n
60 FOR p=0 TO 7
70 POKE d+(256*p),PEEK (c+p)
80 NEXT p
90 NEXT n
100 PAUSE 0
```

Também é possível escrever uma mensagem ao contrário, utilizando-se alguns caracteres que têm funções especiais de controle, como o **volta cursor**, representado por **CHR$ 8**. O programa a seguir realiza esta tarefa.

```
10 LET a$="MICRO SISTEMAS"
20 PRINT AT 10,9;a$;
30 LET v$=CHR$ 8
40 FOR i=1 TO LEN a$
45 PAUSE 60
50 PRINT v$+a$(i)+v$;
60 NEXT i
```

Outros caracteres de controle também podem ser usados livremente, obtendo-se resultados diferentes. Experimente o programa a seguir e tente descobrir como os resultados foram obtidos.

```
10 LET a$=CHR$ 22+CHR$ 10+CHR$
10+"==="
20 LET b$=CHR$ 23+CHR$ 20+"XXX
"
30 LET c$=CHR$ 21+CHR$ 1+"///"
40 PRINT a$+b$
50 PRINT AT 10,10;c$
```

2 – ENTRADA DE DADOS

No TK90X, quando uma instrução INPUT que se refere a uma variável do tipo string é executada, um par de aspas aparece na área de entrada de dados, indicando ao programador que uma variável do tipo string está sendo lida. Porém, é possível controlar este aparecimento de aspas através da instrução **INPUT LINE**. Rode o programa abaixo e perceba a diferença de execução das linhas 10 e 20.

```
  10 INPUT "Digite uma palavra:"
;a$
  20 INPUT "Outra:"; LINE b$
  30 PRINT a$,b$
```

A instrução TAB também poderá ser usada no INPUT. Experimente no programa anterior substituir a linha 20 por **20 INPUT TAB 30;"Outra: "; LINE b$**. Os atributos de COR utilizados com o **PRINT(PAPER,INK,FLASH,BRIGHT)**, também poderão ser usados. Experimente este programa.

```
  10 INPUT INK 3; FLASH 1;"Seu n
ome?";n$
  20 PRINT INK 4;"Bom dia, ";n$
```

3 – UTILIZANDO O JOYSTICK

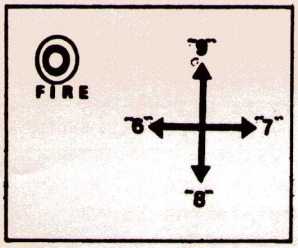


Figura 1

Podemos usar o joystick comum do videogame Atari conectado ao TK90X, obtendo, assim, um ótimo meio de interagir com jogos de ação. A figura 1 mostra a tecla correspondente para cada posição do joystick, e o programa a seguir mostra uma utilização do mesmo.

```
  10 LET a$=INKEY$
  20 IF a$="" THEN GOTO 0
  30 PRINT ("esquerda" AND a$="5
");("direita" AND a$="7");("desc
e" AND a$="8");("sobe" AND a$="9
");("fogo" AND a$="0")
  40 GOTO 10
```

Outro modo de utilizar o joystick (e, por analogia, qualquer tecla!) é através da instrução **IN**. Neste caso, podemos, inclusive, identificar por programa uma direção intermediária. Isto corresponderia, em termos de teclado, a identificar duas teclas pressionadas simultaneamente, o que é altamente desejável ao se programar bons jogos de ação com o uso do joystick, mas não se consegue com a instrução **INKEY$**. Rode o próximo programa e veja os novos resultados. A instrução **IN** é usada neste caso para *ler* o valor de um conjunto de cinco teclas. O endereço 61438, que aparece na linha 10 do programa abaixo, está associado às teclas de 0 a 6.

```
  10 LET s=IN 61438-43
  20 IF s=20 THEN GOTO 10
  30 PRINT ("fogo" AND s=19);("s
obe" AND s=18);("sobe direita" A
ND s=10);("direita" AND s=12);("
desce direita" AND s=8);("desce"
 AND s=16);("desce esquerda" AND
 s=0);("esquerda" AND s=4);("sob
e esquerda" AND s=2)
  40 GOTO 10
```

4 – USANDO VARIÁVEIS DO SISTEMA

Se você quer gerar efeitos não encontrados no BASIC, deixar um programa mais rápido ou torná-lo mais caprichado, a utilização de POKEs na área de variáveis do sistema poderá ajudá-lo e com algumas vantagens. Dentre os endereços e valores existentes, destacamos os seguintes:

a) 23560 - serve para indicar qual tecla foi pressionada (semelhante ao INKEY$). Experimente (e estude) este programinha:

```
10 PAUSE 0: PRINT CHR$ (PEEK 2
3560);: GOTO 10
```

Tecle algo, como se estivesse usando uma máquina de escrever (inclusive com espaço e SHIFT).

b) 23561 - tempo de espera para começar a repetir a tecla que estiver sendo pressionada. O valor mínimo (que dá a maior rapidez) é .5 e o valor normal (quando se liga o micro ou após um NEW) é 35.

c) 23562 - tempo de retardo na repetição de teclas. O valor mínimo é .5 e o normal é 5.

d) 23606 - endereço inicial da tabela de caracteres do micro, que pode
23607 ser modificada à vontade.

e) 23609 - pode-se variar a tonalidade do sinal que é emitido ao se pressionar alguma tecla, colocando neste endereço qualquer valor entre 0 (que é o normal, logo que se liga o micro) até 255.

f) Conjugando duas variáveis, podemos descobrir a quantidade de memória que um programa ocupa. Digite, no modo direto:

```
PRINT (PEEK 23641+256*PEEK 23642
)-(PEEK 23635+256*PEEK 23636)
```

5 – IMPRESSÃO CAPRICHADA

Além dos recursos mencionados no *capítulo 17 do manual do TK90X*, é possível melhorar a impressão no vídeo, utilizando-se das seguintes técnicas:

FLASH - use CAPS SHIFT e SYMBOL SHIFT para obter o cursor E; depois pressione CAPS SHIFT e 9. Tudo o que for escrito agora daqui em diante ficará piscando. Para desativar o FLASH, use (no modo E) as teclas CAPS SHIFT mais 8.

BRIGHT - use CAPS SHIFT e SYMBOL SHIFT para obter o cursor E; depois pressione apenas a tecla 9, sozinha. Agora tudo será escrito com brilho (como BRIGHT 1). Para desativar, pressione apenas a tecla 8, sozinha, após ter entrado no modo E.

PAPER - pressione CAPS SHIFT e SYMBOL SHIFT para obter o modo E; e depois tecle apenas o número correspondente à cor que você deseja para usar no fundo do caráter (PAPER). Para desativar, ponha o microcomputador no modo E e pressione a tecla 7 sozinha.

6 – PROGRAMAS INTERESSANTES

Experimente digitar estes pequenos programinhas, que servirão para demonstrar alguns dos recursos e particularidades do TK90X, além de servir de distração.

```
 10 FOR X=175 TO 0 STEP -1: FOR
Y=0 TO 255: PLOT INK RND*7;Y,X:
NEXT Y: LET X=X-RND*8: NEXT X:
STOP
```

```
 10 FOR X=175 TO 0 STEP -1: FOR
Y=0 TO 255: PLOT INK RND*7;Y,X:
NEXT Y: NEXT X: STOP
```

```
  10 LET A=1: LET B=1: FOR D=1 T
O 185 STEP 5: CIRCLE D,A,B: LET
A=A+2: LET B=B+2: NEXT D: STOP
```

```
  10 INPUT "Entre com um numero
! ";B: CLS : LET A=0: PLOT 0,175
: DRAW 0,-175: PLOT 0,0: DRAW 17
5,0: FOR D=175 TO 0 STEP -B: PLO
T 0,D: DRAW A,-D: LET A=A+B: NEX
T D: GOTO 10
```

7 – SCROLL COM A ROM

Mesmo possuindo um prático recurso de SCROLL para listagem de programas, o TK90X não tem um comando que realize esta função.

Para se obter um SCROLL em seus jogos, basta utilizar a rotina da ROM (a mesma que é usada na listagem dos programas) chamada por **RAND USR 3280**. Teste o seguinte programa-exemplo:

```
  10 PRINT AT 21,9;"Micro Sistem
as"
  20 FOR F=0 TO 20
  30 RAND USR 3280: NEXT F
```

8 – ZERANDO LINHAS

Às vezes, é necessário zerar a primeira linha de um programa para protegê-la contra um **DELETE** indesejável ou simplesmente zerar uma ROM com o nome do autor para dificultar a ação dos piratas. Porém, é muito mais interessante zerar linhas no meio do programa, a fim de dificultar – e muito – a ação dos piratas, vejamos como:

Digite os comandos abaixo numa linha que antecede a que será zerada:

```
 LET A=PEEK 23637+256*PEEK 23638
: POKE A,0: POKE A+1,0
```

e, a seguir, tecle **GOTO** (o número da linha acima).

Por exemplo, no programa abaixo, para zerar a linha 20, digite os comandos na linha 15, depois tecle **GOTO 15** e pronto! Mas não se esqueça de apagar esta linha depois. Agora, tente apagar a linha zerada.

```
  10 PRINT
  15 LET A=PEEK 23637+256*PEEK 2
3638: POKE A,0: POKE A+1,0
  20 PRINT "Linha a ser zerada"
  30 PRINT
```

9 – GRÁFICOS EM ALTA RESOLUÇÃO

À seguir, quatro programas gráficos decorativos muito simples, que podem ser modificados à vontade, podendo ainda ser usados como efeitos visuais em outros programas.

1º) **Rosca** – Este programa pode gerar aleatoriamente até 17 tipos de gráficos (podem ser triângulos, quadrados, estrelas etc.), que, no final, sempre formam uma figura semelhante a uma *rosca*.

```
  10 BRIGHT 1: CLS : FOR A=95 TO
255 STEP 10
```

```
  20 PLOT 75,37: DRAW INK (RND*7
);100,100,A*PI
  30 CLS : NEXT A
```

2º) **Efeitos com DRAW** – Estes dois programas geram lindos gráficos. O primeiro forma uma circunferência e o segundo uma *esteira* que se dobra.

```
  10 BORDER 0: PAPER 0: BRIGHT 1
: OVER 1: CLS
  20 PLOT 65,27: DRAW INK (RND*7
);100,100,PI*255: GOTO 20
```

```
  10 OVER 1: LET X=0: LET X1=255
: LET Y=0: INK (RND*7): CLS
  20 PLOT X,Y: DRAW X1,Y: LET Y=
Y+1: LET X1=X1-6: LET X=X+3: IF
Y(=175 AND X1)-255 THEN GOTO 20
  30 GOTO 10
```

3º) **Retângulos** – Compõe diversos retângulos que formam desenhos harmônicos em toda a tela e depois os apagam.

```
  10 PAPER 6: CLS
  20 LET I=1: LET X=0: LET Y=0:
LET X1=255: LET X2=0: LET X3=-25
5: LET X4=0: LET Y1=0: LET Y2=17
5: LET Y3=0: LET Y4=-175
  30 PLOT X,Y: INK 1: OVER 1: DR
AW X1,Y1: DRAW X2,Y2: DRAW X3,Y3
: DRAW X4,Y4
  40 LET X=X+2: LET Y=Y+2: LET X
1=X1-4: LET X3=X3+4: LET Y2=Y2-4
: LET Y4=Y4+4: IF Y(=175 THEN GO
TO 30: LET I=I+1
  50 GOTO 20
```

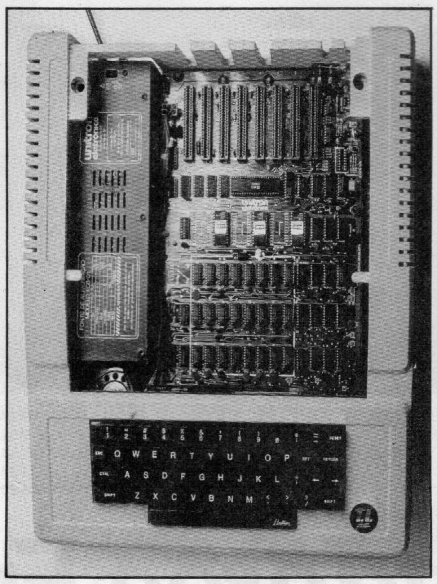
4º) **Quadrado** – Desenha uma forma *quadrada* decorando o seu interior.

```
  10 LET X=0: LET X1=175: OVER 1
  20 PLOT X,0: DRAW INK 1;X1,175
: PLOT 0,X: DRAW INK 1;175,X1
  30 LET X1=X1-8: LET X=X+4: IF
X(=175 THEN GOTO 20
  40 GOTO 10
```

André Koch Zielasko estuda as linguagens BASIC, FORTRAN, COBOL e Assembler, além de pesquisar o emprego dos micros de lógica Sinclair nas áreas de Astronomia e Telecomunicações. André desenvolveu as dicas 4, 5 e 6.

Alexandre Mariano de Brito é estudante e programa em BASIC e Assembler nos microcomputadores TK85 e TK90X, sendo ainda o responsável pelas três últimas dicas.

O UNITRON REVELA O SEU SEGREDO:



MODULARIDADE

Quem compra um micro está investindo no presente e no futuro. Por isso o equipamento deve acompanhar a evolução das necessidades e aplicações do usuário. Através dos seus oito slots (parte superior da foto), o UNITRON APII permite a expansão do equipamento na medida em que crescem as necessidades e de acordo com as economias de cada um.

Você pode começar comprando um UNITRON APII e ligá-lo na TV (P&B ou cores). Depois pode acrescentar disk-drives, impressora, monitor profissional, placa Z-80 para CP/M, placa 80 colunas, interface de comunicações, etc., etc. Você pode também aumentar a memória do seu micro com placas de 16, 32, 64 ou 128K. Tudo isto de modo bastante simples, com manuais em português e fácil instalação. Quanto ao software, você dispõe de programas como o VISICALC, DBASE, WORD STAR e milhares de outros best-sellers, desenvolvidos para o "APPLE II+" norte-americano. Essa modularidade e riqueza de software é que fazem do UNITRON APII o micro mais vendido para empresas, profissionais liberais e famílias inteligentes. Fique com o melhor.

Fique com o UNITRON APII.

O MELHOR CONJUNTO

UNITRON APII, MONITOR COMPO, DRIVE ELEBRA, IMPRESSORA MONICA E INTERFACES.

263 ORTN's

Válido até 28/02/86

assessorarte

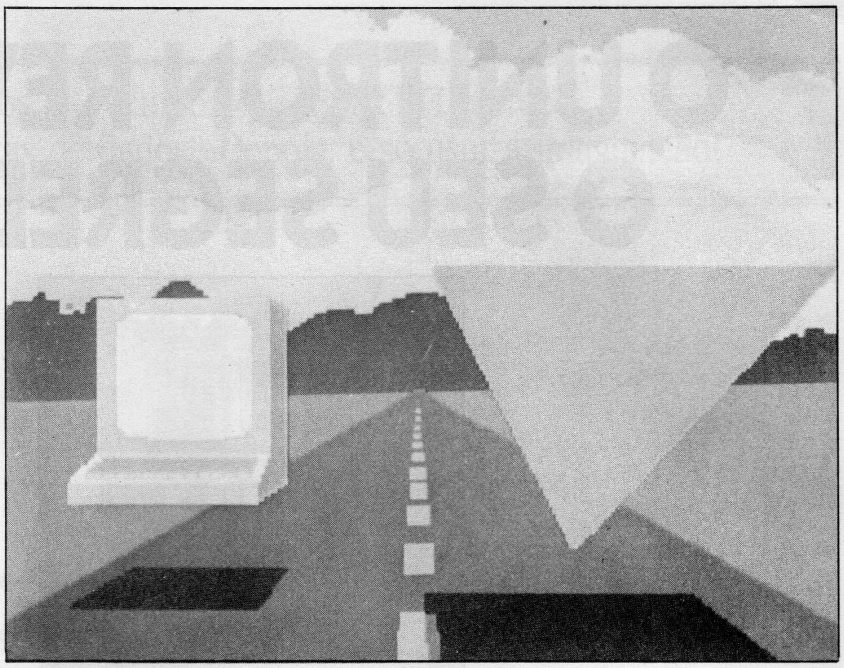
O MAIOR REVENDEDOR UNITRON DE TODO O PAÍS

RUA SETE DE SETEMBRO, 99 - 11º ANDAR
TEL.: PABX (021) 224-7007 - CEP 20050 - RIO DE JANEIRO - RJ

AV. PAULISTA, 726 - 5º ANDAR - CONJ. 506
TEL.: (011) 288-4492 - CEP 01362 - SÃO PAULO - SP

compumicro
INFORMÁTICA EMPRESARIAL LTDA.

Para que você desfrute de todo o potencial da linguagem Forth, qualquer que seja o seu micro, é indispensável conhecer o seu principal instrumento: os compiladores.



As imagens da estrada, do micro e da pirâmide, mostradas na foto, foram geradas por um computador usando o compilador Graforth.

Compiladores Forth

Alta velocidade de processamento, pequeno espaço de memória ocupado e liberdade para criar novas funções e comandos, são algumas das principais características do Forth, uma linguagem que abriu novos horizontes para a utilização do microcomputador nos dias modernos.

Desenvolvida originalmente nos EUA em 1970 para controlar um telescópio, a linguagem Forth em pouco tempo conquistou o seu lugar, principalmente porque, necessitando de pequeno espaço na memória dos computadores, ela se tornou ideal para equipamentos que dispõem de memória reduzida.

É uma linguagem adequada para aplicações em que o alto custo e o espaço ocupado pela memória são essenciais, sendo, por suas características, utilizada na programação de computadores a bordo de naves espaciais, robôs, jogos de vídeo, controle de processos, mecanismos automáticos (desde que não necessitem usar técnicas de inteligência artificial), animação gráfica, tradutores de linguagem portáteis e aplicativos para a área comercial.

OS COMPILADORES

A finalidade básica dos compiladores Forth é, em última análise, converter os programas escritos em Forth para a linguagem de máquina (Assembler) do equipamento onde o programa será rodado. Quanto ao modo de execução, existem dois tipos de compiladores Forth: o alinhavado indireto e o alinhavado por sub-rotina. O primeiro traduz e compacta o código-objeto de um programa para um código intermediário que economiza a memória, porém sacrifica a velocidade; já o segundo, apesar de ocupar maior memória é mais veloz, pois traduz os programas em Forth diretamente para o Assembler.

É preciso observar que todos os compiladores Forth são formados por um núcleo básico (mínimo de rotinas necessárias para a compilação), com tamanho variável de 3 a 8 Kb, responsável pela compilação das rotinas em tempo de execução e a interação com o usuário. Porém, a maioria dos compiladores existentes no mercado ocupa um espaço de memória superior ao do núcleo básico, pois eles normalmente oferecem alguns recursos adicionais. Estes recursos estão armazenados num bloco conhecido como caixa de ferramentas (*Tool Box*), na qual podem estar contidos um editor de textos, pacotes gráficos e/ou rotinas para a manipulação de números fracionários, utilizando a técnica de ponto flutuante.

Opcionalmente, estão disponíveis para ser utilizados com os compiladores Forth, bancos de dados integrados, voltados principalmente para aplicações na área comercial, além de pacotes de programas para a interligação com outros equipamentos que utilizem interfaces compatíveis com os padrões IEE-488 ou RS 232-C.

O MERCADO

Os compiladores Forth existentes no mercado podem ser classificados em duas categorias básicas: simples e gráficos.

Os simples são aqueles que têm como única função: compilar os programas escritos em Forth. Nesta categoria, os principais compiladores disponíveis são o MMS Forth – compilador alinhavado indiretamente para os micros da linha TRS-80, modelos I e III –; ZX Forth – compilador para os micros da linha Sinclair – e o Forth CP/M – compilador adequado a todos os microcomputadores que possam operar sob o CP/M.

Os compiladores gráficos para a linguagem Forth, ao contrário, oferecem diversos recursos extras, como editores gráficos, de texto, sintetizadores de som etc. Eles estão disponíveis para as linhas Apple, Macintosh e IBM-PC.

Para a linha Apple, podem ser encontrados o Graforth – compilador que possui diversos recursos gráficos – e o Transforth – compilador alinhavado por sub-rotina, adequado a aplicações científicas, possibilitando traçar gráficos, criar caracteres em geral (gregos, latinos etc.) e sintetizar sons. O Macintosh, também fabricado pela Apple e ainda não disponível no Brasil, conta com o MacForth, um compilador com um editor de textos, recursos sonoros e gráficos, tendo, inclusive, um

programa que demonstra o funcionamento dos vários componentes (pistões, hastes etc.) de um motor à explosão.

Para os microcomputadores compatíveis com o IBM-PC, existe um compilador alinhavado indireto. Este compilador, porém, é de difícil aquisição no Brasil.

No entanto, tão importante quanto saber o que existe no mercado é ter uma visão clara sobre a utilização dos compiladores Forth existentes. Assim, necessitaremos colher a opinião dos que mais entendem do assunto: os usuários.

OS USUÁRIOS

Eduardo Previdelli é Presidente do Clube de Microusuários, da Sucesu-SP, e trabalha com computação gráfica em microcomputadores. Ele utiliza o Graforth, um compilador-gerenciador de gráficos para os equipamentos da linha Apple, na criação de figuras tridimensionais.

Eduardo trabalha junto à empresa Marcos Briquet Filmes, que também utiliza o Graforth na criação de figuras tridimensionais a serem utilizadas na animação de comerciais e filmes publicitários. Para produzir estes filmes, as imagens geradas na tela do micro são registradas em uma impressora gráfica, sendo em seguida arte-finalizadas (retocadas, pintadas etc.) e por fim filmadas quadro a quadro.

Além do Graforth, Previdelli também possui o Transforth, um compilador mais voltado para aplicações científicas. Porém, até o momento ele não o utilizou por falta de documentação, já que conseguiu uma cópia do programa, mas não têm um manual do mesmo.

Programador de micros há quatro anos, Ivan Camilo da Cruz utiliza o Forth em seu DGT-100 para elaborar programas aplicativos. Segundo Ivan, o que o atraiu para a linguagem Forth foi a possibilidade de criar suas próprias instruções, pois Forth é "uma das poucas linguagens autoconstrutivas, o que permite criar um conjunto de comandos de acordo com as minhas necessidades".

Já o estudante de engenharia civil Thomas Low, autor do compilador Forth para os micros da linha Sinclair, publicado em **MS nº 39**, afirma que a velocidade foi o principal motivo que o levou a escolher o Forth, isto porque é "sete a oito vezes mais rápida que o BASIC". Thomas se entusiasmou tanto que chegou a adaptar este compilador para a utlização com o CP/M 2.2 num microcomputador ED-251.

Também entusiasmado pelo Forth, o Engenheiro de software Cyd Delgado chegou a desenvolver como tese de mestrado, no IME, um compilador Forth que ocupava 7 Kb de memória, adequado para equipamentos que utilizem o CP/M em sua versão 1.4 ou mais recente. Este compilador, segundo Cyd, "foi desenvolvido com recursos adaptados da linguagem MUMPS, visando principalmente aplicações comerciais". Cyd chegou a pensar em comercializá-lo, entretanto, como o programa ainda necessitava de várias modificações para ter condições de ser lançado no mercado – e ele não dispunha de tempo – desistiu da idéia.

Programador e usuário de Forth, Fernando da Costa Grossi trabalha atualmente no Senac Informática e sempre utilizou muito a linguagem Forth por considerar que ela tem todas as vantagens apresentadas pelas outras linguagens e nenhuma desvantagem. "A linguagem Forth é mais rápida, mais compacta e mais flexível", afirma ele. Fernando está atualmente com dois equipamentos diferentes, um TK82 modificado, com 38 Kb de memória RAM, e um TK90X.

Para o primeiro, Fernando utiliza um compilador chamado ZX-Forth, em sua versão original, com um manual traduzido para o português pela Tigre, que adaptou esse compilador e lançou-o sob o nome Tig-Forth. Para utilizar o ZX-Forth, o micro precisa ter apenas 16 Kb de RAM. Já no TK90X, Fernando Grossi utiliza o compilador Fig-Forth, versão original da Abersoft, com manual também original. Neste último, são necessários 48 Kb de memória. Fernando da Costa Grossi afirma ainda que existe pouquíssima documentação sobre a linguagem Forth no Brasil, e recomenda o livro *"Starting Forth"*, que pode ser encontrado em livrarias com setor de informática ou lojas do ramo.

Na opinião de Fernando da Costa Grossi, a linguagem Forth pode ser usada em qualquer aplicação onde se usaria, por exemplo, o BASIC. Ele mesmo já desenvolveu jogos usando Forth e agora está trabalhando no desenvolvimento e estudo de algoritmos. "O Forth é uma linguagem estruturada e muito poderosa", conclui.

Já Milton Maldonado Jr., que trabalhou com Fernando no desenvolvimento de software, afirma que usou a linguagem Forth como experiência e não gostou. "Acho que não é uma linguagem tão poderosa e é difícil", afirma. Quando aos compiladores, Milton cita o Tig-Forth, adaptação da Tigre para a linha Sinclair, e um outro publicado em MICRO SISTEMAS.

Na hora de comprar

Apesar da pouca divulgação que recebem, os compiladores Forth podem ser encontrados com relativa facilidade nas principais lojas e software-houses. Algumas destas empresas como a Soft-Speed, Micromaq e Soft-Line também enviam o disquete ou fita cassete com o programa pelo correio, junto, é claro, com uma cópia do manual do software. Quanto ao manual, é necessário que o usuário possua um mínimo de conhecimento da língua inglesa, pois, raramente o mesmo não está escrito neste idioma. Estão disponíveis, portanto, compiladores para todas as linhas de micros.

Na Micromaq, loja carioca especializada em acessórios e software para os micros da linha TRS-Color, pode ser encontrado o Colorforth (2 ORTN), um compilador que necessita de 16 Kb de memória RAM para operar.

Empresa voltada somente para a área de software, a Softline oferece dois compiladores para a linguagem Forth. O primeiro é o Specforth (1,3 ORTN), um compilador para os micros da linha ZX Spectrum com 48 Kb de RAM. O segundo é o Graforth (4,6 ORTN), compilador adequado aos micros compatíveis com o Apple II.

Já na Login, loja e software-house voltada para os micros da linha TRS-80, encontramos o MMS Forth (3 ORTN) e o Graforth (3 ORTN). Também pode ser adquirido o MVP (2 ORTN), um compilador que pode ser utilizado em todos os micros que operem com o CP/M, a partir da versão 1.4.

Em São Paulo, na Biblioteca Brasileira de Software (BBS), podem ser encontrados os compiladores MMS Forth (3,6 ORTN) da Muller Microcomputer Software e o Graforth (5 a 5,7 ORTN) da Insoft. Além destes, existe um compilador para o CP 400 (1,7 ORTN). Os dois primeiros exigem um drive e também 48 ou 64 Kb de RAM, respectivamente. Já o último precisa de cassete, sendo necessário uma memória de 48 Kb.

A seguir, estão os endereços das lojas consultadas:

Micromaq – *Rua Sete de Setembro, 93 – loja 106, tel: (021) 222-6088, Centro, Rio de Janeiro-RJ.*

Softline – *Av. Graça Aranha, 145 – Slj 01, tel: (021) 262-6968, Centro, Rio de Janeiro-RJ.*

Login – *Av. N. S. de Copacabana, 861 – sala 315, tel: (021) 237-3170, Rio de Janeiro-RJ.*

Softspeed – *Caixa Postal 40088 – Rio de Janeiro-RJ.*

BBS – *Av. Brigadeiro Faria Lima, 1394 – conjunto 82, tel: (011) 210-1251, São Paulo-SP.*

REFERÊNCIAS BIBLIOGRÁFICAS

Para quem deseja saber mais sobre a linguagem Forth, aqui estão algumas sugestões para consultas:

As linguagens de programação, MICRO SISTEMAS nº 19

Forth: uma linguagem rápida e compacta, MICRO SISTEMAS nº 23

Outras palavras em Forth, MICRO SISTEMAS nº 32

Compilador Forth para Sinclair, MICRO SISTEMAS nº 39

Linguagens de Programação: entre o homem e a máquina, MICRO SISTEMAS nº 47

Curso de Forth, MICRO SISTEMAS nº 49, 50, 51, 52 e 53

Reportagem de Carlos Alberto Azevedo e Stela Lachtermacher

Se você é usuário de um micro da linha Apple, saiba como proteger melhor os direitos autorais de seus programas contra os *piratas*.

Protegendo direitos autorais

Clóvis Ático Lima Filho

Certamente o leitor já é usuário de um micro da linha Apple e produz seus próprios programas. É provável, também, que após todos os testes, e uma vez constatado que tudo está justo e perfeito com sua obra, acrescente à mesma algumas linhas REM, com os seus dados pessoais, visando preservar para a posteridade a marca do seu gênio criador.

Infelizmente, porém, linhas de programas são facilmente apagadas ou alteradas e alguns *piratas* sem ética podem simplesmente adulterar os créditos do desprotegido autor, comprometendo seriamente a sua "imortalidade".

Os *piratas* que se cuidem, pois não será mais tão fácil a sua vilania. Doravante, nossos direitos autorais ficarão imortalizados numa linha, cujo número estará "mascarado" sendo, portanto, muito difícil a sua localização e alteração (infelizmente, o impossível não existe neste ramo).

TEORIA

Para os leitores que, como eu, não se conformam em simplesmente copiar uma listagem, aí vão alguns conceitos teóricos.

Provavelmente você já sabe que as linhas dos programas podem ser numeradas de 0 até 63999. O que nem todos sabem é que, na verdade, elas podem ir até 65535; mas isto é assunto para um outro artigo.

Além disso, o amigo também deve saber que uma linha iniciada com a palavra reservada REM é simplesmente ignorada pelo computador. Isto quer dizer que você pode escrever nesta linha tudo que lhe vier na cabeça (resssalvadas as limitações impostas pela censura) sem que o programa seja "gongado" pelo analisador de sintaxe. É aqui que nós vamos trabalhar...

O nosso objetivo, portanto, é camuflar o número da linha que contém os nossos créditos, de modo que o *pirata* não possa identificar o seu verdadeiro valor. Não podendo, assim, deletá-la.

O leitor mais apressado pensará logo em dar POKEs no par de endereço que guarda o número da linha. Fazendo isto, porém, estaremos alterando, de fato, o número da linha, o que não atenderá aos nossos objetivos. Lembre-se que nós queremos esconder, e não alterar, o número da linha.

O grande desafio é fazer com que o cursor, quando ainda na linha, retroceda até o canto esquerdo da tela e recomece a imprimir a partir daí, ou seja, por sobre o verdadeiro número da linha, porém sem apagar nada, senão o analisador de sintaxe "lhe pega".

Ora! Todo mundo sabe que CTRL-H (CHR$ 8) faz justamente isto (da mesma forma que a seta para esquerda), porém se você digitar, por exemplo, uma linha 10 REM e a seguir CTRL-H, o cursor voltará um espaço, resultando em 10 RE, pois esta é a sua função. Se você der um CR (RETURN), obterá um erro de sintaxe quando tentar rodar o programa.

A saída, portanto, tem que ser um pouco mais sutil. Temos que digitar qualquer caráter após o REM e então substituir, no endereço certo, o valor desse caráter por um 8.

Senão, vejamos. Dê um **NEW** e digite a seguinte linha: 10 REMT (sem nenhum espaço). Agora dê o seguinte comando direto: PRINT PEEK (2053).

Você obterá 178, que corresponde à palavra reservada **REM**.

Peça, agora, o **PEEK** (2054) e obterá 84.

Aí está o T que nós vamos substituir...

Se o leitor der, agora, um **POKE** 2054,8, o T será substituído por um CTRL-H. Liste a linha 10 para confirmar. Nós estamos no caminho certo.

Se você entendeu até agora, nós podemos fazer um exemplo prático. Digite a seguinte linha: 10 REMTTTT... (não digite nenhum espaço depois do **REM**).

Agora peça os PEEKs de 2054 a 2057 e você obterá 84 em todos eles (ou seja, quatro Ts), e de 2058 a 2060 você obterá 46 (que são os três pontinhos).

Dê os seguintes comandos diretos:

POKE 2054,8
POKE 2055,8
POKE 2056,8
POKE 2057,8

Digite então **LIST**. Você deverá obter: 10...

O que aconteceu?

O cursor imprimiu 10 REM mais um espaço, retrocedeu quatro espaços, devido aos quatro CTRL-H, e imprimiu, começando no lugar do R, os três pontinhos. Deu prá perceber?

Tente agora o seguinte:
10 REMTTTTTTTT..... (oito Ts e sete pontos) e POKE, de 2054 a 2061,8.

Se você fez tudo certinho, deverá obter, quando listar, apenas sete pontinhos na tela, ou seja, sumiu também o número da linha.

Os curiosos podem diminuir a velocidade do micro, com um **SPEED = 50**, por exemplo, e assistir à proeza em Slow Motion (não esqueça de retornar à velocidade normal, após isto).

Já deu prá perceber que, ao invés de pontinhos, você poderá escrever o seu nome após o último T, que ele ficará preservado dos "piratas" menos experientes.

Supondo que o seu programa comece na linha 100, você poderá camuflar o seu nome em qualquer linha entre 0 e 99, pois uma vez que esta passará a ser a primeira linha do programa, os endereços são os mesmos que vimos. Considerando, porém, que existem "piratas" muito espertos por aí, precisamos agir com mais cautela, sendo uma boa idéia exibir os seus direitos autoriais no final do programa, ou, quem sabe, no meio dele, onde será bem mais difícil localizá-los.

Você deve estar pensando como irá descobrir os endereços onde estarão guardados os Ts, para poder substituí-los.

Isto seria um trabalho que exigiria toda a paciência do notório Jó e que, provavelmente, resultaria em fracasso. Vamos deixar esta mão-de-obra para o próprio micro...

PRÁTICA

O truque será colocar na linha REM caracteres identificadores, ou seja, caracteres que possam ser localizados pelo computador em meio a todos os outros, sem o perigo de confundi-lo.

Os caracteres de número 33 (SHIFT 1 = !) e 38 (SHIFT 6 = &) são pouco usuais e deverão servir para isso, mas na realidade você poderá usar a maioria dos caracteres, desde que eles não apareçam em nenhuma parte do seu programa (o T usado na teoria já não serve mais).

Agora, você escolhe um número cabalístico qualquer, entre 0 e 63999, para ser o número verdadeiro da sua linha REM (lembre-se que este número não poderá ter sido usado no seu programa, pois ele é verdadeiro). Agora escolha um outro número, mais cabalístico ainda, para servir de número falso. Use a sua fértil imaginação; pode ser um número já existente no seu programa, um inteiro possível, um número decimal, uma expressão matemática, uma mistura de letras e números, letras, sinais, espaços em branco, enfim, qualquer coisa que lhe venha na cabeça, pois a loucura que você inventar vai ser agora impressa no lugar do número verdadeiro.

Digite, então, a sua linha, incluindo após o **REM** tantos caracteres identificadores quantos forem necessários e, a partir daí, a seqüência de caracteres que irá esconder o número da linha.

Uma dica importante: a quantidade de caracteres identificadores é função da quantidade de dígitos do número verdadeiro da linha. Use a regrinha: número de caracteres identificadores = 6 + quantidade de dígitos do número da linha.

Agora, o "pulo do gato". Faça o computador pesquisar todos os valores do seu programa, entre os quais se encontra a sua linha REM, e mande-o substituir todos os caracteres identificadores que encontrar pelo caminho por um CTRL-H. Fácil, não é?

Não esqueça de avisá-lo que número de linha e apontadores de próxima linha não valem.

Vamos fazer um exemplo: Carregue um programa qualquer de sua própria produção (pois, imagino que você não é um "pirata sem ética").

Vamos, então, criar uma linha **REM** qualquer que, apenas a título de exemplo, fique como última linha do programa. Não esqueça que ela deve ter um número pouco usual, como 61234. Este número tem cinco dígitos, o que significa dizer que iremos precisar de 6+5 =11 caracteres identificadores.

Digite, então:
61234 REM!!!!!!!!!! FOI FEITO POR MIM

Feito? Agora, lá no início do programa, digite (supondo que seu programa começe na linha 100):

```
10 E =  PEEK (175) + 256 *  PEEK
    (176) - 2
20  FOR W = 2053 TO E
30  IF  PEEK (W) = 0 THEN W = W +
    3: GOTO 60
40  IF  PEEK (W) = 33 THEN  POKE
    W,8
60  NEXT
80  END
```

Confira direitinho e mande rodar. Você terá que aguardar alguns segundos, em função do tamanho do seu programa. Quando o cursor aparecer, dê um **LIST**.

Se tudo deu certinho, no lugar da sua linha 61234 deverá haver simplesmente FOI FEITO POR MIM.

Só para alimentar seu ego, dê um LIST 61234.

Beleza, não é? Considerando que apenas você sabe o número da linha, será muito difícil localizá-la.

Para encerrar o trabalho, é só deletar as linhas de 10 a 80 e gravar o programa com seus direitos autorais protegidos, mas não o faça ainda.

Vamos fazer um trabalho definitivo, usando agora dois caracteres identificadores, visando substituir o segundo por um Retorno de Carro (CR), ou seja, CHR$ 13.

Delete a linha 61234 e digite uma nova linha com o seguinte formato (as únicas coisas que não podem ser modificadas são as ! e os & , o resto fica a seu critério):

```
XXXXX  REM !!!!!!!!!!YYYYY&=====
     ==========&NOME DO PROGRAM
     A&&COPYRIGHT 1985 BY&SEU NOM
     E&CIDADE - ESTADO&==========
     =======
```

Onde XXXXX é o número verdadeiro da linha e YYYYY é o número falso. Agora, acrescente à primeira rotina as seguintes linhas:

```
50  IF  PEEK (W) = 38 THEN  POKE
    W,13
70  DEL 10,70
```

E mande rodar...

Agora Liste a sua obra. Satisfeito?

O seu programa, assim, tem uma linha **REM** de aspecto profissional, cujo número verdadeiro somente o criador, o seu micro e você mesmo sabem qual é.

Querem saber o que foi feito? Então lá vai...

Na linha 10 foi definida uma variável (E), cujo valor corresponde ao fim do seu programa, não importando o tamanho dele.

A linha 20 abriu um laço, que vai do início do programa (2053) até o fim dele (E).

A linha 30 é uma armadilha para final de linha, que impede que coincidências desagradáveis modifiquem os apontadores de linha seguinte e os números das linhas do programa.

As linhas 40 e 50 ficam encarregadas de, uma vez achados os caracteres identificadores, substitui-los adequadamente.

A linha 60 dispensa explicações e a linha 70 provoca um suicídio coletivo.

A linha 80 interrompe a rotina, após a sua conclusão, se não o seu programa sairia rodando.

O único defeito do seu programa, agora, é que ele inicia em uma linha 80 END e, portanto, não roda.

Delete esta linha, grave o seu programa e vá dormir tranqüilo, pois você acaba de entrar para o restrito "Clube de Imortais".

Clóvis Ático Lima Filho é Geólogo e trabalha no Departamento Nacional de Produção Mineral, onde desenvolve programas para as áreas de Geologia, Engenharia e Economia num Exato MC-4000. É também usuário de um TK85.

bits

MSX evolui

Apesar do sucesso alcançado em todo o mundo, os fabricantes de microcomputadores padrão MSX não se acomodaram e têm procurado aprimorar cada vez mais o seu produto, oferecendo equipamentos mais sofisticados e a preços mais acessíveis.

Visando aumentar a versatilidade destes equipamentos, recentemente novos acessórios foram desenvolvidos ou adaptados para o padrão MSX. Já estão disponíveis no mercado europeu, por exemplo, joysticks sensíveis ao toque ou que operam por infravermelho, mouses e tabletes gráficos para desenho. Para o armazenamento de dados, também já estão disponíveis vários gravadores especialmente projetados. Dentre eles, podemos citar o Sanyo DR202, um gravador que localiza automaticamente qualquer programa numa fita cassete, carregando o programa diretamente para o microcomputador, sem necessidade de intervenção manual.

Alguns fabricantes que adotaram o padrão MSX também não se restringiram às características básicas ao desenvolverem os seus equipamentos. A Yamaha, por exemplo, um dos maiores fabricantes mundiais em instrumentos musicais eletrônicos, dotou o seu microcomputador CX-5M com características sonoras superiores às dos outros equipamentos, possibilitando, dentre outras coisas a conexão de um teclado musical eletrônico. O CX-5M possui ainda recursos de solo e até 46 vozes, que abrangem desde o som de sinos até o ruído do vento.

Porém, a mais recente novidade é o surgimento de uma nova série denominada MSX 2.0 que, entretanto, ainda mantém a principal característica do padrão MSX: total compatibilidade a nível de software.

Nesta série foram implementadas as seguintes modificações: fixação do número de colunas em 80 (não mais em 40); aumento da ROM onde está contido o BASIC para 64 Kb; aumento da resolução gráfica para 512 x 424 pontos; possibilidade para ampliar a memória de vídeo para 128 Kb; incorporação de um novo circuito integrado gerador de vídeo (o V/9938) com 10 modos gráficos; saída para impressora paralela; e entrada para mouse e dois joysticks.

Quanto aos periféricos, além dos já existentes, os micros do padrão MSX versão 2.0 poderão utilizar também um novo acessório: o digitalizador de imagens. Com ele é possível combinar figuras produzidas pelo computador com outras provenientes de uma câmera ou vídeocassete, possibilitando a produção e edição de um filme, incluindo a colocação de títulos e legendas.

SAVE

A partir de agora, as agências de viagens passam a contar com mais um instrumento para agilizar os seus serviços. Trata-se do SAVE – Sistema de Automação de Dados para Agências de Viagens e Turismo, desenvolvido pela Medidata em conjunto com a Datamension Informática, para utilização nas agências da empresa Bel-Air Viagens e Turismo.

Desenvolvido na linguagem MUMPS, o SAVE é composto por vários módulos, que podem efetuar automaticamente a reserva de passagens aéreas, hotéis, veículos, etc. Atualmente o SAVE já está operando nos computadores M301, M2001 e M3001 da Medidata, estando prevista para agosto a apresentação do sistema completo já em funcionamento no supermicro M1001, também na Medidata.

Nova fábrica da Moore

A Moore Formulários investiu Cr$ 9 bilhões em sua fábrica, em Gravataí, no Rio Grande do Sul. Esta é a quinta fábrica da empresa, numa iniciativa de regionalização, para que os centros produtores fiquem mais próximos dos consumidores, reduzindo assim o tempo de entrega e o preço dos formulários contínuos e padronizados. A nova fábrica da Moore emprega modernas técnicas de temperatura e controle de umidade, proporcionando às áreas de produção a mesma temperatura dos CPDs, onde os formulários serão utilizados mais tarde.

Magnex terá 16 bits

A Magnex espera triplicar o seu faturamento em 1986. A receita para isso será o lançamento de novos equipamentos. Ela ingressará no mercado de 16 bits, com a produção do M 160, pretendendo lançar no segundo semestre uma nova linha para empresas de maior porte.

Novos programas

A Micropic, representante da US Computadores, lançou 100 novos programas. Dentre eles aplicativos para 16 bits PC e Xt, como clínica médica, controle de estoque, contabilidade, mala direta, controle de Loto e sistemas de custo, com preços a partir de 100 ORTN. Para a linha Apple, estão sendo comercializados controle de estoques, representações, comércio de telefones, dentista, cadastro para gado, folha de pagamento, banco de dados com processador de texto e mala direta. O telefone da Micropic é (011) 283-6200 - SP.

FENASI 86

Confirmando o desenvolvimento do setor de suprimentos para informática, está programada para os dias 24 a 27 de abril no Palácio das Convenções do Anhembi em São Paulo, a abertura da 1ª Feira Nacional de Acessórios, Suprimentos e Instalações para Informática (I FENASI).

Reunindo aproximadamente 70 empresas do setor, o evento tem o apoio da SEI e ANFORSAI (Associação Nacional dos Fornecedores de Suprimentos e Acessórios para Informática), além de contar com a presença de empresas como a IBM, Burroughs, Moore, Texprint, Data Ribbon e Vector.

Cartuchos de impressão

Visando proporcionar uma grande durabilidade e alta qualidade de impressão, a Scritta Eletrônica está produzindo uma linha completa de cartuchos para as impressoras Grafix e Epson. Disponível para os modelos RC-80 (para a Grafix 80 e 80F/T) e RC-100 (para a Grafix 100 e 100HS), estes cartuchos permitem a impressão na cor preta de até 3×10^6 caracteres.

Maiores informações podem ser obtidas na Scritta, situada à Alameda Amazonas, 832, Alphaville, Barueri, São Paulo - SP, tel.: (011) 421-3422.

Mudanças na Brascom

A Brascom, agora com nova diretoria, já entrou com pedido de levantamento de concordata, o que deverá ocorrer dentro em breve, segundo informou seu Diretor Comercial Fernando Vieira Guimarães.

Após a fase de revisão e estudos dos projetos em andamento, a Brascom decidiu ampliar a performance de seu sistema em cerca de 30%, sem implicar em mudanças estruturais. A empresa tomou esta decisão preocupada em manter a compatibilidade dos mais de 1000 equipamentos instalados em todo o país.

Preço alterado

O redefinidor de caracteres, para micros compatíveis com o ZX-81 (analisado na Seção Hardware do nº 51 de MS), está sendo, na verdade, comercializado por 5 ORTN. Esta informação foi dada pelo próprio fabricante, que tem sede na Praia de Icaraí, 211 - loja 03, Niterói - RJ, tel.: (021) 717-1700.

Acessórios para desenho

O CPD da revista MICRO SISTEMAS recebeu recentemente dois acessórios sob medida para aqueles que desejam aproveitar os recursos gráficos das linhas Apple e TRS Color: o Compopad e o Colormax. Fabricado pela Compo do Brasil, empresa conhecida no mercado pela fabricação de monitores de vídeo, o Compopad é a versão nacional do Koala Pad. Consiste num traçador gráfico (prancheta com estilete) e no programa Compo Micro Ilustrador, que permite a criação e edição das telas criadas com o manuseio da prancheta. O Colormax é um pacote composto de uma placa e um software, que possibilita trabalhar a parte gráfica do Color através de "pull-down menus" e interessantes comandos de pincel, borracha e zoom. É fabricado pela empresa carioca Novatec Indústria e Comércio, situada na Rua Paissandu, 346, tel.: (021) 245-5823.

SOFT SPEED

GARANTIA TOTAL • MANUAIS INCLUSOS

JOGOS: ESCOLHA 3 E PAGUE 2 !!! E AINDA TEM BRINDE

CP-400 JOGOS EM FITA

F101 - DECATHLON - O jogo mais famoso de todos
F102 - DOUBLE BECK - Prenda as imagens
F103 - DRACONIAN - Salve o astronauta no espaço
F104 - ICE HOCKY - Hockey no gelo
F105 - MR. DIG - Fantastico e emocionante
F106 - INTERCEPT 4 - Destrua os inimigos galácticos
F107 - MISTAR - Jogo de aventura
F108 - SIMULADOR DE VÔO - Excelente qualidade gráfica
F109 - POPEYE - Derrote o Brutus
F110 - GALAGON - Fantastico jogo!
F111 - ZAXXON - O melhor jogo para micros
F112 - GERM - Defenda-se dos germes
F113 - DEVIL ASSAULT - Emocionante!
F114 - AIR TRAFFIC CONTROL - Controle o aeroporto
F115 - CRASH - Defenda-se dos insetos
F116 - SPACE WRECK - Defencia sua base espacial
F117 - SHAFT - Destrua os pontos e muito cuidado!
F118 - DESERT RIDER - Corrida de carros no deserto
F119 - CHOPPER STRIKE - Aventura com helicóptero
F120 - TIME PATROL - Salve os pára-quedistas

PREÇO DE CADA JOGO: Cr$ 35.000
PARA CADA 2 JOGOS COMPRADOS, ESCOLHA UM TERCEIRO INTEIRAMENTE GRÁTIS. E AINDA GANHE MAIS UM JOGO: BLACKJACK.

CP-400 APLICATIVOS E UTILITÁRIOS EM DISCO

D121 - VIPCALC Planilha eletrônica	250.000
D122 - VIPWRITER - Processador de textos	250.000
D123 - VIPDATABASE - Banco de Dados	250.000
D124 - VIPTERM - Transforma o micro em terminal	250.000
COLEÇÃO VIP (4 PROGRAMAS ACIMA)	600.000
D125 - PRO COLOR FILE - Banco de dados	200.000
D126 - ELITE FILE - Banco de dados	200.000
D127 - ELITE CALC - Planilha eletrônica	200.000
COLEÇÃO ELITE (2 PROGRAMAS ACIMA)	300.000
D128 - DYNACALC - Planilha Eletrônica	200.000
D129 - DYNAGRAPH - Utilitário para gráficos	200.000
COLEÇÃO DYNA (2 PROGRAMAS ACIMA)	300.000
D130 - TELE WRITER 64 - Processador de textos	200.000
D131 - GRAPHICON COMPLETO - Super útil. gráfico	400.000
D132 - SPEED-CONTABILIDADE COMPLETA	400.000
D133 - SPEED-CADASTRO DE CLIENTES	400.000
D134 - SPEED-FOLHA DE PAGAMENTO	400.000
D135 - SPEED-CONTROLE DE ESTOQUES	400.000
COLEÇÃO SPEED (4 PROGRAMAS ACIMA)	1.000.000
D136 - KIT EDTASM E MACRO80C - Assemb/Disassem	200.000
D137 - KIT SYNTHER 77 E MUSIC II - Som/Música	200.000
D138 - PASCAL - Linguagem científica	300.000
D139 - FORTRAN - Linguagem científica	300.000
D140 - FORTH - Linguagem científica	300.000
COLEÇÃO DAS TRÊS LINGUAGENS ACIMA	700.000
D141 - PIRATECTOR - Proteje seu programa contra pirataria e coloca capas apresentação	400.000
D142 - CLONE - Copia discos protegidos	400.000
D143 - CLONE MASTER - O melhor copiador discos	500.000
D144 - COLOR KIT - Copia disco/fita e fita/disco	300.000
D145 - TAPEUTIL - Copia disco/fita e fita/disco	300.000
D146 - DISK UTILITY - Copia, back-up, diretório, endereços programas, lista arquivos, etc.	200.000
D147 - DISK DRIVER ANALYSER - Analisa hardware	200.000
D148 - COLOR QUAVER - Super sintetizador de som	300.000
D149 - TITULADOR PARA VT - Gerador caracteres	300.000

BRINDE: NA COMPRA DE QUALQUER PROGRAMA EM DISCO PARA CP400 **GRÁTIS** UM SUPER SINTETIZADOR DE VOZ.

CP400 UTILITÁRIOS EM FITA

F150 - SUPER SCREEN 51 × - 24 Linhas texto/graf	60.000

JOGOS EM FITA PARA TK90X

F201 - XADREZ - Em sua melhor versão
F202 - REVERSI - Famoso jogo Othelo
F203 - GAMÃO - Jogo de tabuleiro
F204 - ANT ATTACK - Destrua as formigas
F205 - CYBER RATS - Cuidado com os ratos!
F206 - HORÁCIO E AS ARANHAS - Um grande jogo!
F207 - 3D TANX - Batalha com tanques
F208 - SIMULADOR DE VÔO - Impressionante!
F209 - COOKIE - Ajude o cozinheio
F210 - HORÁCIO ESQUIANDO - Aventura divertida
F211 - PYJAMARAMA - Tente vencer este desafio
F212 - ANDROID 2 - Sensacional!
F213 - PLANETOIDS - Excitante missão espacial
F214 - MISSILE - Cuidado! míssil fatal
F215 - PSST - Destrua os insetos
F216 - MANIC MINER - Conduza Willy nas cavernas
F217 - HUNGRY HORACE - Cuidado com os quadros
F218 - PINBALL - Tradicional flipper
F219 - SPACE RAIDERS - Defenda seu planeta
F220 - ASTRO BLASTER - Cuidado com os invasores
F221 - JET SET WILLY - Tente fazer Willy dormir
F222 - GHOST HUNT - Sensacional versão do Pacman
F223 - CHEQUERED FLAG - Pilote um F1 de verdade
F224 - TENNIS - Excelente versão
F225 - ROBOTICS - Destrua os robos invasores!
F226 - GALACTIC ABDUCTORS - Cuidado! perigo!
F227 - SINUCA - Tente matar todas as bolas
F228 - MINED OUT - Cuidado com as minas
F229 - MR. WIMPY - Pegue os ingredientes
F230 - GOLFE - Excelente e real jogo
F231 - CROSS - Tente atravessar a rua perigosa
F232 - JUMPING JACK - Cuidado com as fendas no chão
F233 - GROUND ATTACK - Penetre na caverna inimiga!
F234 - MONSTER MINE - Cuidado com os monstros da mina
F235 - SEIDDAB ATTACK - Proteja sua cidade das aves
F236 - GALAXXIANS - Destrua os invasores

PREÇO DE CADA JOGO: Cr$ 35.000
PARA CADA 2 JOGOS COMPRADOS, ESCOLHA UM TERCEIRO INTEIRAMENTE GRÁTIS. E AINDA GANHE MAIS UM JOGO: SENHA.

APPLE II - JOGOS ESPECIAIS EM DISCO - CADA CR$ 120.000

D301 - KARATECA - Lute karate de verdade
D302 - CONAN - Aventura igual ao filme
D303 - GHOSTBUSTERS - Aventura igual ao filme
D304 - SUMMER GAMES - 8 Modalidades esportivas
D305 - CPI - Jogo pornográfico para adultos
D306 - SKYFOX - Pilote um caça de verdade

APPLE II - JOGOS EM DISCO DA PROMOÇÃO

D307 - AZTEC - Decifre a pirâmide azteca
D308 - PICADILLY - Sensacional jogo
D309 - SHIFT SAM - Fantástico e emocionante jogo
D310 - DONKEY KONG - Mário contra o gorila
D311 - FREE FALL - Tente vencer este desafio
D312 - PIEMAN - Excelente e divertido
D313 - ORBITRON - Entre em órbita com seu micro
D314 - Q*BERT - Tente se movimentar nos cubos
D315 - JUNGLE HUNT - Aventura na selva
D316 - CAVERNS OF CALISTO - Aventura
D317 - PITFALL II - Nas cavernas perdidas
D318 - TAXMAN - Labirinto tipo Pacman
D319 - XADREZ - Tradicional jogo
D320 - GAMÃO - Jogo milenar de tabuleiro
D321 - BUCK ROGERS - Pilote uma espaçonave
D322 - LOAD RUNNER - Pegue os tesouros

PREÇO DE CADA JOGO: Cr$ 65.000
PARA CADA 2 JOGOS COMPRADOS, ESCOLHA UM TERCEIRO INTEIRAMENTE GRÁTIS. E AINDA GANHE MAIS UM JOGO: POKER

APPLE II - UTILITÁRIOS E APLICATIVOS EM DISCO

D351 - DISK DRIVE DIAGNÓSTICO	200.000
D352 - SAY IT - Sintetizador de Voz	200.000
D353 - MUSICOMP - Componha sua Música	200.000
D354 - KOALA - Criador de Desenhos	200.000
D355 - PRINT SHOP - Gerador de Impressos	170.000
D356 - THE COMPLETE GRAPHICS SYSTEM	200.000
D357 - SCREEN WRITER - Processador de Textos	200.000
D358 - SPEED-MALA DIRETA	300.000
D359 - SPEED-CONTROLE DE ESTOQUES	300.000
D360 - SPEED-CONTAS A RECEBER	300.000
D361 - SPEED-FOLHA DE PAGAMENTO	300.000
D362 - LOCKSMITH 5.0 - Utilitário de Cópia	170.000
D363 - MAGIC WINDOW - Processador de Textos	300.000
D364 - VISICALC II - Planilha Eletrônica	170.000
D365 - VISIFILE II - Banco de Dados	170.000
D366 - DBASE II - Completo	500.000
D367 - FRIDAY - Completo	450.000
D368 - WORDSTAR - Processador de Textos	400.000
D369 - LOTERIA ESPORTIVA	170.000
D370 - LOTO	170.000

BRINDE: GRÁTIS, NA COMPRA DE QUALQUER PROGRAMA EM DISCO UM CURSO PASSO-A-PASSO EM BASIC.

COMO COMPRAR:

NÃO RECORTE SUA REVISTA. FAÇA O SEU PEDIDO POR CARTA, RELACIONANDO SEUS DADOS E OS CÓDIGOS DOS PROGRAMAS DESEJADOS. ANEXE À CARTA UM CHEQUE NOMINAL À ATI Editora Ltda., Av. Pres. Wilson, 165 grupo 1210 - Centro, CEP 20.030 - Rio de Janeiro - RJ. As despesas do correio já estão incluídas.

SOFTSPEED - CAIXA POSTAL 40088 - RIO-RJ

---

rm EDITORA

NOVOS LANÇAMENTOS

OS 40 MELHORES JOGOS PARA SEU TK/CP200 e SINCLAIR — Uma sensacional coletânea de jogos de inteligência e ação, incluindo jogos em linguagem de máquina. Código 200.1. Preço: Cr$ 39.990,

INTRODUÇÃO BÁSICA AOS MICROCOMPUTADORES E À INFORMÁTICA, de Marcelo Meirelles e Renato Audi. Livro que aborda de forma simples e clara a evolução dos computadores, linguagens de programação, atividades profissionais, entre outros assuntos. Código 300.2. Preço: Cr$ 39.990,

NAS MELHORES LIVRARIAS DO PAÍS

PEDIDOS PARA ATI Editora Ltda., Av. Pres. Wilson, 165 grupo 1210 CEP 20.030 - Rio de Janeiro - RJ, anexando cheque nominal.

BRASIL TRADE CENTER

Comércio, Indústria e Participações S.A.

COMPUTER SHOPPING

Mesas para Microcomputadores

BTC-05 M

Cr$ 1.849.000

BTC-03 M

Cr$ 1.139.000

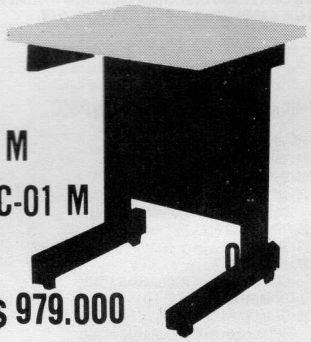
Fabricação própria ● Cores discretas ● Desenho moderno

BTC-02 M

Cr$ 1.359.000

BTC-04 M

Cr$ 1.349.000



BTC-01 M

Cr$ 979.000

EQUIPAMENTOS

Micros das linhas: Sinclair ● TRS-80 ● CCE ● Unitron ● Dismac ● TK 2000 II ● TK 85 ● Exato Pro ● Impressoras Modem ● Etc.

SUPRIMENTOS

Periféricos ● Disketes ● Fitas ● Drives ● Placas de Expansão ● Interfaces ● Formulário Contínuo ● Cabos ● Arquivos para Disketes ● Etc.

SOFTWARE

Programas e Jogos variados para todas as linhas e marcas de Computadores

VIDEOS

Curso de Inglês em vídeo cassete ● Telão ● Suporte para TV ● Transcodificação para todos os sistemas ● Fitas: VHS - BETA-U-MATIC e para limpeza de cabeça ● Baterias para 2 e 8 hs. ● Iluminadores ● Cabos de extensão para Câmaras ● Bolsas para Câmaras e Vídeos

Nas lojas Brasil Trade Center, você também encontra vídeo-game, cartucho com jogos, som, telefonia, etc., das melhores marcas.

VISITE UMA DE NOSSAS LOJAS E COMPROVE O QUE ANUNCIAMOS

Matriz: Av. Epitácio Pessoa, 280
Ipanema-RJ — Tels.: 259-1299/259-1499

Filiais: Rua da Assembléia, 10 — Loja S-112
Centro-RJ — Tel.: 222-5343

Rua Silva Vale, 416 — Cavalcante-RJ
Tels.: 592-3047/592-3098

Rua Lopes Trovão, 134/Sala 201
Niterói-RJ — Tel.: 710-3659

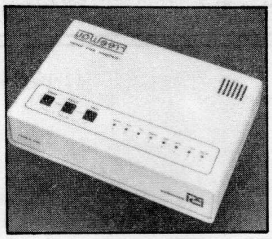
Rua Conde de Bonfim, 229 — Loja A
Tijuca-RJ — Tel.: 284-2031

SERVIÇO EXPRESSO REMETEMOS PARA TODO O BRASIL

OFERTA VÁLIDA COM A APRESENTAÇÃO DESTE ANÚNCIO

Freeway, uma interface inteligente

A PSI – Projetos e Serviços em Informática investiu cerca de Cr$ 300 milhões no desenvolvimento do Freeway, um periférico para todos os tipos de microcomputadores, que atua como interface entre micro e impressora. Interface inteligente, o Freeway otimiza o uso do equipamento ao armazenar informações velozmente, liberando-o para outras atividades. Possui memória real de 64 Kb, expansível até 512 Kb, mas pode operar com o dobro desta capacidade porque trabalha com armazenamento em modo comprimido. Ele engloba funções como fornecer estatísticas de impressão, permitir avanço de páginas, inserir textos em meio a grandes relatórios, imprimir gráficos. Além disso, controla a troca de papel avulso, converte modos de comunicação serial em paralelo e vice-versa, libera rapidamente linhas telefônicas em transmissão de listagens a grande distância e realiza auto-testes.



Freeway, um novo periférico.

Apresentado em gabinete leve, com tamanho reduzido, o Freeway está à venda, ao preço base de 90 ORTN, na PSI, à Avenida Pavão, 346, São Paulo, telefone 533-0112, ou em seus revendedores.

Catálogo Micromaq

A Micromaq – loja carioca especializada em software e acessórios para os micros da linha TRS Color – está lançando o seu catálogo de produtos.

Com 24 páginas ilustradas, o catálogo descreve toda a linha de produtos oferecidos pela empresa, com os respectivos preços em ORTN.

Para obter o catálogo da Micromaq, os interessados devem solicitar por carta, telefone ou pessoalmente na loja, situada à Rua Sete de Setembro, 92 - loja 106, CEP 20050, Rio de Janeiro - RJ, tel.: (021) 222-6088.

Sorteio TK90X

Os vencedores da terceira e última rodada do concurso "Ganhe um TK90X", promovido pela Microdigital em conjunto com MICRO SISTEMAS, são Daltiva Carl Fortes, de Passo Fundo, Rio Grande do Sul; e João Egidio Romão Jr., de São Paulo, Capital.

Assessoria Total

Oferecendo assistência técnica e assessoria a nível de software e hardware para os micros das linhas IBM-PC e Apple, além de comercializar equipamentos da Microtec, Unitron, CCE, Elebra, Olivetti, Grafix e Elgin, a Computerware está completando dois anos de atividades.

Agora com uma nova loja no Infoshopping e o atendimento especializado de vendedores e técnicos, a empresa espera conquistar uma fatia ainda maior do mercado constituído pelas pequenas e médias empresas, profissionais liberais e executivos.

A Computerware fica à Av. Almte. Barroso, 91/sala 1108, tel.: (021) 240-7294, Rio de Janeiro - RJ. O show-room do Infoshopping fica à Rua do Catete, 311/loja 107.

Grupo Machline

A Sid Informática encerrou o ano de 85 com um faturamento de mais de 700 bilhões de cruzeiros, o que representou um crescimento real (descontado a inflação) de 21% em relação ao ano de 84.

Em termos de novos produtos, a Sid já anunciou o lançamento do PC-Júnior, uma versão do PC Sid 501 voltada para aplicações em teleprocessamento; o segundo minicomputador da série H; um PC-AT uma nova versão do terminal Sid 1700, através do qual se obtém informações e pode-se fazer saques; e ainda um novo terminal de vídeo para uso com os minis Sid.

Já a Sharp, empresa do mesmo grupo, deverá fechar o exercício 85/86 (que se encerra em 30 de março próximo), com faturamento bruto superior a Cr$ 2 trilhões, o que representa um crescimento real de 30%.

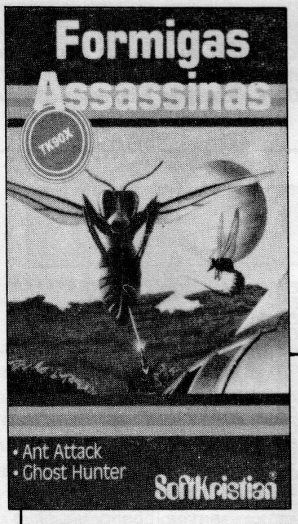
Segundo o Presidente da Divisão Sharp, Daniel Dazcal, o novo micro da empresa – Hot Bit – poderá representar cerca de 10% do faturamento de 86. Até o final do primeiro semestre deverá estar no mercado o drive para esse equipamento.

PC/PTE

A empresa ADR – Applied Data Research lançou recentemente no mercado americano o software PC/PTE, um pacote que permite a todos os micros compatíveis com o IBM-PC (inclusive as versões XT e AT), acessar arquivos em mainframes – computadores de grande porte.

O PC/PTE executa os comandos externos do PC DOS e aceita as estruturas de diretória do DOS 2.0, permitindo ao programador manipular os dados, editar os códigos e transferir os arquivos para o mainframe.

O PC/PTE ainda não está disponível no Brasil, porém o seu representante exclusivo, a SCI – Sistemas Computação e Informática, promete para breve a chegada deste software.



Novos jogos

Os usuários dos micros compatíveis com o TK90X, já podem contar com mais 36 jogos para a sua diversão.

Desenvolvidos pela Softkristian, estes programas estão agrupados em 18 fitas cassete, cada uma contendo dois jogos. As fitas são Cookin, Horácio, Formigas Assassinas, Super Simulador, Space Raiders, Planetoids, Pole Position, Super Tênis, Pyjamarama, Pssst, Manic Miner, Jet Pac, Galatic, Batalha dos Tanques, Willy, Jumping Jack, Flash Attack, Xadrez e Gamão.

O preço de cada fita é 0,6 ORTN e elas podem ser encontradas nas lojas especializadas, magazines e lojas de departamentos.

Cursos & Seminários

RJ-A PC Software programou, para o mês de fevereiro, os seguintes cursos: A-B-C (de 27 a 28), Wordstar (de 24 a 26), dBase III (de 11 a 14 e de 18 a 21), Lotus 1-2-3 (de 10 a 13 e de 18 a 21) e Symphony (de 11 a 14 e de 17 a 20). Maiores informações podem ser obtidas na PC Software, situada à Av. Almirante Barroso, 91, sala 415, Centro, tel.: (021) 220-5371.

RJ-A Datamicro Informática está oferecendo para o mês de março dois cursos. O primeiro é o "micro-mulher" (5,2 ORTN), um curso de introdução à microinformática para mulheres com início em 5/03, já o segundo é o "Mumps para mini e micros" (15 ORTN), com início previsto para 6/03. O endereço da Datamicro é Rua Visconde de Pirajá, 547, sala 211, Ipanema, tel.: (021) 511-0395.

RJ-O Instituto de Tecnologia ORT está oferecendo os seguintes cursos: Atualização em Microinformática (de 17/02 a 19/02), Introdução à Informática e Microcomputadores (de 24/02 a 27/02) e Planilhas Eletrônicas: Visicalc e Supercalc (de 03/03 a 11/03). Todos os cursos podem ser feitos nos horários de 15:00 às 18:00h ou de 19:00 às 22:00h. Maiores informações podem ser obtidas na Secretaria do Centro de Informática, através do tel.: (021) 286-7842, ramal 75.

RS-Robótica, CAD/CAM, CIM, CPL's e Sistemas flexíveis de manufatura, são alguns dos temas que serão debatidos no Seminário sobre Automação Industrial para a Região Sul, promovido pela SOBRACON – Sociedade Brasileira de Controle Numérico, para os dias 16 e 17 de abril em Porto Alegre. Maiores informações podem ser obtidas pelos telefones: (011) 255-2967 SP, (0512) 24-0053 RS e (0482) 33-3597 SC.

SP-A SCI – Sistemas Computação e Informática programou para os dias 19 a 21 deste mês no Hotel Ca d'Oro, o curso Microcomputadores na Empresa, destinado a executivos, administradores, gerentes e profissionais liberais. Maiores informações podem ser obtidas na própria SCI, situada a Av. Paulista, 949 - 11º andar, tel.: (011) 289-0099.

SP-A RB Consultoria em Sistemas realizará de 25/02 a 26/02, o curso Microinformática no Planejamento Financeiro.

Maiores informações poderão ser conseguidas na RB, à Rua Luiz Coelho, 308 - 5º andar, conjunto 53, tels.: (011) 256-1007 e 259-3149.

Utilizando este programa e um micro da linha ZX81, você resolverá os seus problemas de fatorial, arranjo e combinação.

Arranjo no Micro

Marcelo Contart

Apesar dos micros de lógica Sinclair serem considerados como "videogames", porque a maior parte do software existente no País é de jogos, estes micros possuem uma excelente capacidade para realizar cálculos numéricos, como qualquer outro de maior porte. Para explorar essa capacidade, é necessário que se faça uma rotina adequada, requerendo do programador apenas um pouco de criatividade.

Com este programa, você poderá resolver problemas de Fatorial, Arranjo e Combinação em poucos segundos, mas, em primeiro lugar, vamos ter uma "aulinha" para relembrar a teoria.

FATORIAL

Chama-se "n fatorial" ou "fatorial de n" (e se indica por "n!") o produto de n números inteiros. Exemplos:

Fatorial de 4 = 4! = 4 * 3 * 2 * 1 = 24

Fatorial de 5 = 5! = 5 * 4 * 3 * 2 * 1 = 120

ARRANJO

Um agrupamento é considerado um arranjo quando não podemos trocar a ordem de seus termos, pois desta forma será mudado o agrupamento. Usamos a seguinte fórmula para o cálculo do arranjo:

$$A_{n,k} = \frac{n!}{(n-k)!}$$

Se desejarmos agrupar seis letras (A,E,I,O,F,L) de quatro em quatro, formaremos diversas palavras, sendo que, se trocarmos as letras de lugar, será formada uma palavra diferente da primeira (por exemplo, FILA é diferente de LIFA). Exemplo:

Arranjo de seis letras agrupadas quatro a quatro:

$$A_{6,4} = \frac{6}{(6-4)!} = \frac{6!}{2!} = 360$$

COMBINAÇÃO

Um agrupamento é considerado combinação quando podemos trocar a ordem de seus termos à vontade, não alterando, em conseqüência, o agrupamento.

Usamos a seguinte fórmula para o cálculo da combinação:

$$C_{n,k} = \frac{n!}{(n-k)!k!}$$

Se tivermos que agrupar cinco pessoas (A,B,C,D,E), formando duplas, observaremos que a dupla AB será a mesma dupla que BA, e, portanto, para calcular o número de combinações possíveis para cinco pessoas agrupadas duas a duas, faremos:

$$C_{5,2} = \frac{5!}{(5-2)!2!} = \frac{5!}{3!2!} = 10$$

O PROGRAMA

As linhas de 1 a 105 são somente para apresentação, vindo em seguida o menu (linhas 110 a 290). Após o menu, você tem três opções, ou seja, o cálculo do fatorial (linhas 500 a 680), o cálculo do arranjo (linhas 1000 a 1290) ou o cálculo da combinação (linhas 1500 a 1800).

Se você quiser rodar o programa, não digite **CONT**, e sim **RUN**.

Para gravar o programa em fita cassete, basta digitar **GOTO 8000**; e para recuperá-lo, use **LOAD ""** ou **LOAD "F.A.C. (MC).**".

DOIS KBYTES

Se o seu microcomputador possui somente dois Kb de memória, proceda da seguinte maneira para poder utilizar este programa:

1º) Elimine todas as linhas menores que 110, exceto as linhas 16 e 17.

2º) Toda linha que contiver a instrução **GOSUB 5000** deve ser substituída pela instrução **SCROLL**, devendo-se também eliminar as linhas 5000, 5100 e 5200.

3º) Coloque todos os números dentro da função **VAL**. Exemplo:

Antes: 230 FOR F = 1 TO 3

Depois: 230 FOR F = VAL "1" TO VAL "3"

Marcelo Contart tem 18 anos e está cursando o segundo período do curso de Administração de Empresas na UFRJ. É proprietário de um CP-200 e desenvolve programas educativos.

Listagem 1

```
  1 REM ************************
  2 REM
  3 REM AUTOR: MARCELO CONTART
  4 REM
  5 REM * PARA INFORMACOES E
  6 REM . TROCA DE PROGRAMAS:
  7 REM * AV.DR.SOARES DE
  8 REM   OLIVEIRA, 179
  9 REM * 14500-ITUVERAVA/SP
 10 REM * FONE: (016)729-2440
 11 REM
 12 REM DIGITADO POR: BR
 13 REM
 14 REM ************************
 15 REM MICRO SISTEMAS-MAIO/84
 16 LET X$="ENTRE COM O VALOR D
E ""N"""
 17 LET Y$="ENTRE COM O VALOR D
E ""K"""
 18 LET A$="          A R R E J O
 N T A          "
 19 LET B$=CHR$ 128
 20 LET A$=A$+CHR$ 151+B$+CHR$
178+B$+CHR$ 166+B$+CHR$ 183+B$+C
HR$ 168+B$+CHR$ 170+B$+CHR$ 177+
B$+CHR$ 180+B$
 21 LET A$=A$+B$+CHR$ 168+B$+CH
R$ 180+B$+CHR$ 179+B$+CHR$ 185+B
$+CHR$ 166+B$+CHR$ 183+B$+CHR$ 1
85+B$+CHR$ 151
 25 FOR F=20 TO 0 STEP -1
```

```
  30 PRINT AT F,0;A$
  35 NEXT F
  40 FOR F=0 TO 21
  50 IF F=21 THEN SCROLL
  60 PRINT ASN F,7;A$(8 TO 26)
  70 IF F=21 THEN GOTO 110
  80 FOR G=0 TO 3
  90 NEXT G
 100 PRINT AT F,0;A$(33 TO )
 105 NEXT F
 110 GOSUB 5000
 120 PRINT "  FATORIAL"
 130 GOSUB 5000
 140 PRINT "              ARRANJO"
 150 GOSUB 5000
 160 PRINT "                    C
OMBINACAO"
 170 GOSUB 5000
 180 PRINT " DIGITE ""F"" PARA F
ATORIAL,"
 190 GOSUB 5000
 200 PRINT " DIGITE ""A"" PARA A
RRANJO,"
 210 GOSUB 5000
 220 PRINT " DIGITE ""C"" PARA C
OMBINACAO."
 230 FOR F=1 TO 3
 240 SCROLL
 250 NEXT F
 260 IF INKEY$="F" THEN GOTO 500
 270 IF INKEY$="A" THEN GOTO 100
0
 280 IF INKEY$="C" THEN GOTO 150
0
 290 GOTO 260
 499 REM CALCULO DO FATORIAL
 500 GOSUB 5000
 510 PRINT "          CALCULO DO FA
TORIAL"
 520 GOSUB 5000
 530 GOSUB 6000
 540 LET MC=550
 550 GOSUB 5000
 560 PRINT "ENTRE COM UM NUMERO
(""N"")";
 570 INPUT N
 575 IF N=1000 THEN GOTO 110
 580 IF N<0 THEN GOTO 570
 590 IF N>33 THEN GOTO 7500
 600 IF N<>INT N THEN GOTO 7000
 605 PRINT "=";N
 610 LET C=N
 615 REM ROTINA PARA O CALCULO D
O FATORIAL
 620 FOR F=N TO 2 STEP -1
 630 LET N=N*(F-1)
 640 NEXT F
 650 IF C=0 THEN LET N=1
 660 SCROLL
 670 PRINT "FATORIAL DE ";C;" =
";N
 680 GOTO MC
 999 REM CALCULO DO ARRANJO
1000 GOSUB 5000
1010 PRINT "          CALCULO DO AR
RANJO"
1020 GOSUB 5000
1030 GOSUB 6000
1040 LET MC=1100
1050 GOSUB 5000
1060 PRINT "CONSIDERANDO ARRANJO
 DE"
1070 SCROLL
1080 PRINT """N"" TERMOS AGRUPAD
OS ""K"" A ""K"""
1100 GOSUB 5000
1110 PRINT X$;
1120 INPUT A1
1125 IF A1=1000 THEN GOTO 110
1130 IF A1<1 THEN GOTO 1120
1135 PRINT "=";A1
1140 SCROLL
1150 PRINT Y$;
1160 INPUT B1
1165 IF B1<1 THEN GOTO 1160
1170 PRINT "=";B1
1180 IF A1<>INT A1 OR B1<>INT B1
 THEN GOTO 7000
1185 IF A1<B1 THEN GOTO MC
1190 LET A2=A1
1210 REM ROTINA PARA O CALCULO D
O ARRANJO
1220 FOR F=A1 TO A1-B1+2 STEP -1
1230 LET A1=A1*(F-1)
1240 IF A1>1E35 THEN GOTO 7500
1250 NEXT F
1260 SCROLL
1270 PRINT "AR.DE.";A2;" ";B1;"A
";B1;" =";A1
1290 GOTO MC
1499 REM CALCULO DA COMBINACAO
1500 GOSUB 5000
1510 PRINT "          CALCULO DA CO
MBINACAO"
1520 GOSUB 5000
1530 GOSUB 6000
1540 LET MC=1590
1550 GOSUB 5000
1560 PRINT "CONSIDERANDO COMBINA
CAO DE"
1570 SCROLL
1580 PRINT """N"" TERMOS AGRUPAD
OS ""K"" A ""K"""
1590 GOSUB 5000
1600 PRINT X$;
1610 INPUT I1
1615 IF I1=1000 THEN GOTO 110
1620 IF I1<1 THEN GOTO 1610
1625 PRINT "=";I1
1630 SCROLL
1640 PRINT Y$;
1650 INPUT G1
1655 IF G1<1 THEN GOTO 1650
1660 PRINT "=";G1
1670 IF I1<>INT I1 OR G1<>INT G1
 THEN GOTO 7000
1680 LET I2=I1
1690 LET G2=G1
1700 REM ROTINA PARA O CALCULO D
A COMBINACAO.
1710 FOR F=I1 TO I1-G1+2 STEP -1
1720 LET I1=I1*(F-1)
1730 IF I1>1E35 THEN GOTO 7500
1740 NEXT F
1750 FOR F=G1 TO 2 STEP -1
1760 LET G1=G1*(F-1)
1770 NEXT F
1780 SCROLL
1790 PRINT "CO.DE ";I2;" ";G2;"A
";G2;" = ";I1/G1
1800 GOTO MC
5000 SCROLL
5100 SCROLL
5200 RETURN
6000 GOSUB 5000
6010 PRINT "  LEMBRETE "
6020 SCROLL
6030 PRINT "PARA VOLTAR AS INSTR
UCOES"
6040 SCROLL
6050 PRINT "DIGITE ""N"" =1000"
6060 GOSUB 5000
6070 RETURN
7000 GOSUB 5000
7010 PRINT "SOMENTE NUMEROS INTE
IROS."
7020 GOTO MC
7500 GOSUB 5000
7510 PRINT "NUMERO MUITO GRANDE.
"
7520 GOTO MC
8000 SAVE "F.A.C.(MC)."
8010 RUN
```

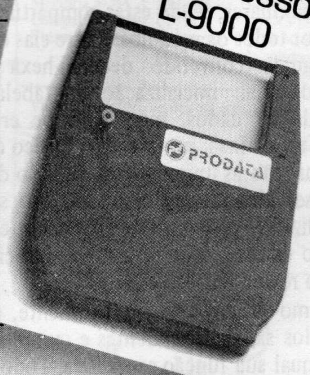

Suprimentos Prodata
uma boa impressão do que foi gravado.

LANÇAMENTO
Fita impressora MX-80

Agora suas impressoras **MX-80, Grafix-80, Prológica P-500, ISA/Ecodata,** e similares já podem contar c/ a boa impressão das fitas **Prodata.**

Consulte-nos sobre preços especiais de lançamento.

LANÇAMENTO
Fita impressora L-9000



PRODATA

Rua Henrique Ongari, 103 CEP 05038 — S.Paulo — Tels.: 864-8599 (linha tronco)

Representantes: Rio de Janeiro: fone-253.3481/Belo Horizonte: fones-225.9871 e 225.4235/ Brasilia:fone-223.6293/Curitiba:fone-263.3224/Porto Alegre:fones-26.6063 e 26.1319/Recife:fone-227.2969.

Aqui está um fácil e eficiente utilitário, escrito em BASIC, para editar, debugar, ler e gravar rotinas em Assembler Z-80. Aproveite.

Sistema Assembler

Helvécio Carneiro R. Netto

Este programa se propõe a ser um eficiente utilitário para editar, debugar, ler e gravar rotinas em Assembler Z-80, ou seja, tudo o que os outros utilitários que existem no mercado fazem, com uma única diferença: ele é essencialmente fácil. É fácil de entender, de operar, de modificar e, além de tudo, escrito em BASIC.

O sistema é composto de módulos distintos e totalmente independentes. Tal característica é que dá ao programa a qualidade de ser facilmente alterável. Se o leitor achar um dos modos de operação dispensável, pode simplesmente abandoná-lo, sem prejuízo para os outros.

Além dos modos de operação (**A, B, C, D** e **E**), o sistema é composto por rotinas auxiliares, sendo estas compartilhadas por todos os módulos. Entre elas encontramos conversão decimal-hexa e hexa-decimal, inicialização de tabelas, entrada de dados, mensagens de erro etc. Tais rotinas, junto com o núcleo do sistema, são as únicas partes que não devem ser alteradas pelo usuário. Ao ser inicializado, o sistema requisita um endereço de carga, que é o endereço inicial da sua rotina Assembler.

Como já foi dito anteriormente, os módulos são independentes e cumprem cada qual sua função específica. Todo o sistema gira em torno de um único estado: o estado **OK**. É dele que partem as ordens e é a ele que retornam os modos de operação, como está ilustrado na figura 1. Para abandonarmos a tarefa e retornarmos ao Monitor, selecionamos o modo **FIM**.

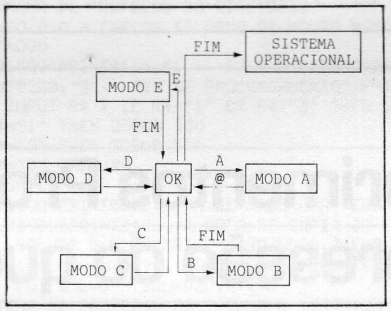


Figura 1: Organização do sistema

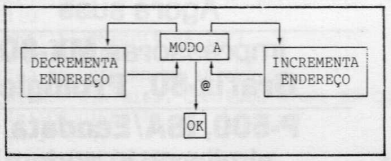
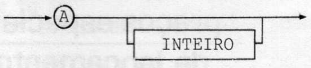


Figura 2: A flexibilidade do modo A

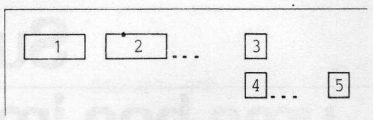
MODOS DE OPERAÇÃO

O modo **A** tem a seguinte sintaxe:



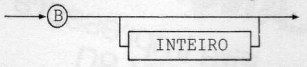
Ela seleciona o display da memória a partir do endereço especificado (o default será o endereço de carga) e admite avançarmos ou retrocedermos na memória, o que permite grande flexibilidade no trato com o conteúdo dos endereços, conforme demonstra a figura 2. Após a emissão do primeiro relatório, o módulo esperará até apertarmos uma das três teclas-comandos – ponto e vírgula (;), traço (–) e arroba ( ) – para em seguida executar a ordem correspondente.

O modo **A** gera o seguinte relatório de saída:



onde: **1** é o endereço em decimal; **2**, o endereço em hexa; **3**, o código decimal do conteúdo da memória; **4**, o código hexa do conteúdo da memória; e **5**, o caráter correspondente ao código ASCII do conteúdo da memória.

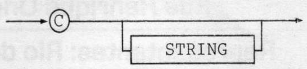
A sintaxe do modo **B** é a seguinte:



Ela altera o conteúdo de endereços de memória a partir do endereço especificado (o default será o endereço de carga). Este modo **B** vem substituir a pseudo-instrução **DEFB**, com a vantagem de mostrar o caráter correspondente ao novo conteúdo da memória.

O modo **B** gera um relatório de saída equivalente ao do modo **A**, sendo que os itens **1** e **2** são mostrados antes de entrarmos com o novo conteúdo da memória, que ocupará então o item **3**. Para retornarmos ao estado **OK**, entramos com o comando **FIM**, quando aparecer o cursor.

Já o modo **C** tem a seguinte sintaxe:

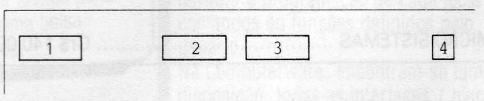


Sistema Assembler

```
5 CLEAR 1000
10 ON ERROR GOTO 3000:DEFINT I,E,J:CLS:PRINT"SISTEMA ASSEMBLER Z
-80":GOSUB 1100
20 PRINT"ENDERECO INICIAL DE CARGA ... ";:GOSUB 1000:GOSUB 1500:
E1=VAL(X$):E=E1:IF E1(=0,20
30 GOSUB 1900
40 PRINT"
OK":GOSUB 1000:IF X$="",40 ELSE GOSUB 1500:IF X$="FIM",END
50 A$=LEFT$(X$,1):IFA$)"@",ON ASC(A$)-64 GOTO 80,60,120,170,190
55 ER=1:GOSUB 2000:GOTO 40
60 IF LEN(X$)=1,E2=E ELSE X$=RIGHT$(X$,LEN(X$)-1):E2=VAL(X$):IF
E2(=0,ER=4:GOSUB 2000:GOTO 40
70 PRINT E2;" ";:E1=E2:J=3:GOSUB 1600:PRINT X$;" ... ",:GOSUB 10
00:IF X$="FIM",40 ELSE GOSUB 1700:IF A=-1,ER=2:GOSUB 2000:GOTO 7
0 ELSE PRINT,,E1;" ... ";CHR$(E1):POKE E2,E1:E2=E2+1:GOTO 70
80 IF LEN(X$)=1,E2=E ELSE X$=RIGHT$(X$,LEN(X$)-1):E2=VAL(X$):IF
E2(=0,ER=4:GOSUB 2000:GOTO 40
90 PRINT E2;" ";:E1=E2:J=3:GOSUB 1600:PRINT X$;" ... ",:E1=PEEK
(E2):J=1:GOSUB 1600:PRINT X$:PRINT,,PEEK(E2);" ... ";CHR$(PEEK(E
2))
100 A$=INKEY$:IF A$="",100 ELSE IF A$="@",40 ELSE IF A$="-",E2=E
2-1 ELSE IF A$=";",E2=E2+1 ELSE 100
110 GOTO 90
120 IF LEN(X$)=1,C$="Z-80"ELSE C$=RIGHT$(X$,LEN(X$)-1):J=1
130 PRINT,"ENDERECO INICIAL ... ";:GOSUB 1000:GOSUB 1500:IF VAL(
X$)(E,130 ELSE E2=VAL(X$):PRINT,"ENDERECO FINAL ... ";:GOSUB 1
000:GOSUB 1500:IF VAL(X$)(E,130 ELSE E3=VAL(X$)
140 CLS:PRINT C$;" ";E2;" ";E3;:PRINT#-1,C$,E2,E3
150 J=1:FOR I0=E2 TO E3 STEP 127:A$="":FOR J0=0 TO 126:E1=PEEK(I
0+J0):GOSUB 1600:A$=A$+X$:PRINT@62,X$:IF J0+I0=E3,PRINT#-1,A$:GO
TO 40 ELSE NEXT
160 PRINT#-1,A$:NEXT:GOTO 40
170 IF LEN(X$))1,ER=1:GOSUB 2000:GOTO 40 ELSE INPUT#-1,X$,E2,E3:
CLS:PRINT X$;" ";E2;" ";E3;:E1=E2:GOSUB 1900:E=E2
180 C=0:FOR I0=E2 TO E3 STEP 127:INPUT#-1,A$:FOR J0=1 TO LEN(A$)
STEP 2:X$=MID$(A$,J0,2):PRINT@62,X$;:GOSUB 1700:POKE I0+C,E1:C=C+
1:NEXT:NEXT:GOTO 40
190 IF LEN(X$)=1,E2=E ELSE X$=RIGHT$(X$,LEN(X$)-1):E2=VAL(X$):IF
E2(E,ER=4:GOSUB 2000:GOTO 40
200 PRINT E2;" ";:E1=E2:J=3:GOSUB 1600:PRINT X$;" ... ",:GOSUB 1
000:GOSUB 1500:IF X$="FIM",40 ELSE IF LEN(X$)(3,IF X$()"EI"AND X
$()"DI",200
205 GOSUB2100:IFA=-1,200
210 FOR I=1 TO 256:IF X$=TO$(I),I=I-1:POKE E2,I:J=1:E1=I:GOSUB 1
600:PRINT,X$:E2=E2+1:GOTO 200
220 NEXT
225 FOR I=1 TO 23
230 A=LEN(X$)-2:IF LEFT$(X$,A)=LEFT$(T1$(I),A) AND (MID$(T1$(I),
A+1,1)=" "OR MID$(X$,7,1)="","),GOSUB 2200:GOTO 200
240 NEXT
250 IF LEFT$(X$,3)="OUT",IF RIGHT$(X$,2)=",A",POKE E2,211:C$=X$:
E1=211:J=1:GOSUB 1600:PRINT,X$;" ";:X$=MID$(C$,4,2):PRINT X$:
GOSUB 1700:IF A=-1,ER=2:GOSUB 2000:GOTO 200 ELSE E2=E2+1:POKE E2
,E1:E2=E2+1:GOTO 200 ELSE 500
260 FOR I=1 TO 22:A=LEN(X$)-4:IF LEFT$(X$,A)=LEFT$(T3$(I),A)AND(
MID$(T3$(I),A+1,1)=" " OR MID$(X$,7,1)="","),GOSUB 1800:GOTO 200
270 NEXT
280 IF LEFT$(X$,3)="LD(" AND MID$(X$,8,2)=","),",A=4:IF RIGHT$(X$,
2)="HL",E1=34 ELSE IF RIGHT$(X$,1)="A",E1=50 ELSE 500 ELSE 300
290 POKE E2,E1:J=1:C$=MID$(X$,A,4):GOSUB 1600:PRINT,X$;" ";:X
$=RIGHT$(C$,2):PRINT X$;" ";:GOSUB 1700
295 IF A=-1,ER=2:GOSUB 2000:GOTO 200 ELSE E2=E2+1:POKE E2,E1:X$=
LEFT$(C$,2):PRINT X$:GOSUB 1700:IF A=-1,E2=E2-1:ER=2:GOSUB 2000:
GOTO 200 ELSE E2=E2+1:POKE E2,E1:E2=E2+1:GOTO 200
300 IF LEFT$(X$,6)="LDHL,(" AND RIGHT$(X$,1)=")",E1=42:A=7:GOTO
290
310 IF LEFT$(X$,5)="LDA,(" AND RIGHT$(X$,1)=")",E1=58:A=6:GOTO 2
90
500 ER=3:GOSUB 2000:GOTO 200
1000 X$="":PRINT CHR$(14D);
1010 A$=INKEY$:IF A$="",1010
1020 IF A$=CHR$(13),PRINT CHR$(8):RETURN ELSE IFA$=CHR$(8) AND L
EN(X$)()0,X$=LEFT$(X$,LEN(X$)-1)ELSE IF LEN(X$+A$)(255,X$=X$+A$
1030 PRINT CHR$(8);A$;CHR$(14D);:GOTO 1010
1100 DIM TO$(256)
1110 FOR I=1 TO 256:READ TO$(I):NEXT
1111 DATA"NOP","-","LD(BC),A","INCBC","INCB","DECB","-","RLCA","
EXAF,AF'","ADDHL,BC","LDA,(BC)","DECBC","INCC","DECC","-","RRCA"
,"-","-","LD(DE),A","INCDE","INCD","DECD","-","RLA","-","ADDHL,D
E","LDA,(DE)","DECDE","INCE","DECE","-","RRA","-"
1112 DATA"-","-","INCHL","INCH","DECH","-","DAA","-","ADDHL,HL",
"-","DECHL","INCL","DECL","-","CPL","-","-","-","INCSP","INC(HL)
","DEC(HL)","-","SCF","-","ADDHL,SP","-","DECSP","INCA","DECA","
-","CCF","LDB,B","LDB,C","LDB,D","LDB,E","LDB,H"
1113 DATA"LDB,L","LDB,(HL)","LDB,A","LDC,B","LDC,C","LDC,D","LDC
,E","LDC,H","LDC,L","LDC,(HL)","LDC,A","LDD,B","LDD,C","LDD,D","
LDD,E","LDD,H","LDD,L","LDD,(HL)","LDD,A","LDE,B","LDE,C","LDE,D
","LDE,E","LDE,H","LDE,L","LDE,(HL)","LDE,A"
1114 DATA"LDH,B","LDH,C","LDH,D","LDH,E","LDH,H","LDH,L","LDH,(H
L)","LDH,A","LDL,B","LDL,C","LDL,D","LDL,E","LDL,H","LDL,L","LDL
,(HL)","LDL,A","LD(HL),B","LD(HL),C","LD(HL),D","LD(HL),E","LD(H
L),H","LD(HL),L","HALT","LD(HL),A"
1115 DATA"LDA,B","LDA,C","LDA,D","LDA,E","LDA,H","LDA,L","LDA,(H
L)","LDA,A","ADDA,B","ADDA,C","ADDA,D","ADDA,E","ADDA,H","ADDA,L
","ADDA,(HL)","ADDA,A","ADCA,B","ADCA,C","ADCA,D","ADCA,E","ADCA
,H","ADCA,L","ADCA,(HL)","ADCA,A","SUBB","SUBC"
1116 DATA"SUBD","SUBE","SUBH","SUBL","SUB(HL)","SUBA","SBCA,B","
SBCA,C","SBCA,D","SBCA,E","SBCA,H","SBCA,L","SBCA,(HL)","SBCA,A"
,"ANDB","ANDC","ANDD","ANDE","ANDH","ANDL","AND(HL)","ANDA","XOR
B","XORC","XORD","XORE","XORH","XORL","XOR(HL)"
1117 DATA"XORA","ORB","ORC","ORD","ORE","ORH","ORL","OR(HL)","OR
A","CPB","CPC","CPD","CPE","CPH","CPL","CP(HL)","CPA","RETNZ","P
OPBC","-","-","-","-","PUSHBC","-","RSTO","RETZ","RET","-","-","-","
-","-","RST8","RETNC","POPDE","-","-","-"
1118 DATA"PUSHDE","-","RST16","RETC","EXX","-","-","-","-","-","
RST24","RETPO","POPHL","-","EX(SP),HL","-","PUSHHL","-","RST32",
"RETPE","JP(HL)","-","EXDE,HL","-","-","-","RST40","RETP","POPAF
","-","DI","-","PUSHAF","-","RST48","RETM","LDSP,HL","-"
1119 DATA"EI","-","-","-","RST56"
1120 DIM T1$(23):FOR I=1 TO 23:READ T1$(I):NEXT
1121 DATA"LDB, 6","LDC, 14","DJNZ 16","LDD, 22","LDE,
30","LDH, 38","LDL, 46","LD(HL),54","LDA, 62","ADDA, 198"
,"ADCA, 206","INA, 219","SBCA, 222"
1131 DATA"SUB 214","AND 230","XOR 238","OR 246","CP
254","JR 24","JRNZ, 32","JRZ, 40","JRNC, 48","JRC,
56"
1140 DIM T3$(22):FOR I=1 TO 22:READ T3$(I):NEXT
1141 DATA"LDBC, 1","LDDE, 17","LDHL, 33","LDSP, 49","JPNZ,
194","CALLNZ,196","JPZ, 202","CALLZ, 204","JPNC, 210","CALLNC
,212","JPC, 218","CALLC, 220","JPPO, 226","CALLPO,228","JPPE,
234"
1142 DATA"CALLPE,236","JPP, 242","CALLP, 244","JPM, 250","CA
LLM, 252","JP 195","CALL 205"
1160 DIM TE$(4):FOR I=1 TO 4:READ TE$(I):NEXT
1161 DATA"LDI160","LDIR176","LDD168","LDDR184"
1200 RETURN
1500 A$=X$:X$=""
1510 FOR I=1 TO LEN(A$):IF MID$(A$,I,1)=" ",NEXT ELSEX$=X$+MID$(
A$,I,1):NEXT
1520 RETURN
1600 X$=""
1610 FOR I=J TO 0 STEP -1:A=INT(E1/(16[I)):IF A)9,X$=X$+CHR$(55+
A)ELSE X$=X$+CHR$(48+A)
1620 E1=E1-A*16[I:NEXT
1630 RETURN
1700 E1=0:IF LEN(X$)()2,A=-1:RETURN
1710 FOR I=1 TO 0 STEP -1:A=ASC(MID$(X$,2-I,1)):IF A)57 AND A(65
,A=-1:RETURN
1720 IF A)57,A=A-55 ELSE A=A-48
1730 IF A(0 OR A)15,A=-1:RETURN ELSE E1=E1+A*16[I:NEXT
1740 RETURN
1800 C$=RIGHT$(X$,4):X$=RIGHT$(T3$(I),LEN(T3$(I))-7):E1=VAL(X$):
POKE E2,E1:J=1:GOSUB 1600:PRINT,X$;" ";RIGHT$(C$,2);" ";LE
FT$(C$,2)
1810 X$=RIGHT$(C$,2):GOSUB 1700:IF A=-1,ER=2:GOSUB 2080:GOTO 200
ELSE E2=E2+1:POKE E2,E1:X$=LEFT$(C$,2):GOSUB 1700:IF A=-1;ER=2:
GOSUB 2000:E2=E2-1:GOTO 200 ELSE E2=E2+1:POKE E2,E1:E2=E2+1
1820 RETURN
1900 J=3:GOSUB 1600:A$=LEFT$(X$,2):X$=RIGHT$(X$,2):GOSUB 1700:PO
KE 16526,E1:X$=A$:GOSUB 1700:POKE 16527,E1
1910 RETURN
2000 PRINT:ON ER GOTO 2010,2020,2030,2040
2010 PRINT" COMANDO INEXISTENTE ":RETURN
2020 PRINT" CONSTANTE HEXA INVALIDA ":RETURN
2030 PRINT" MNEMONICO NAO RECONHECIDO ":RETURN
2040 PRINT" ENDERECO INVALIDO ":RETURN
2100 FOR I=1 TO 4:IF X$=LEFT$(TE$(I),LEN(X$)),POKE E2,237:E2=E2+
1:PRINT,"ED ";:X$=RIGHT$(TE$(I),3):E1=VAL X$):POKE E2,E1:E2=E
2+1:J=1:GOSUB 1600:PRINT X$:A=-1:RETURN
2110 NEXT:A=0:RETURN
2200 C$=RIGHT$(X$,2):X$=RIGHT$(T1$(I),LEN(T1$(I))-7):E1=VAL(X$)
2210 POKE E2,E1:J=1:GOSUB 1600:PRINT,X$;" ";C$:X$=C$:GOSUB 17
00
2220 IF A=-1,ER=2:GOSUB 2000 ELSE E2=E2+1:POKE E2,E1:E2=E2+1
2230 RETURN
3000 PRINT"ERRO N#";ERR
3010 IF E=0,RESUME 20 ELSE RESUME 40
```

Ela cria um arquivo em fita com os códigos hexa dos conteúdos da memória, indagando um endereço inicial e um endereço final, obrigatoriamente maior que o anterior. O arquivo criado tem o nome dado pela expressão *string*, sendo válidos quaisquer caracteres. O default cria um arquivo de nome **Z-80**. Cada registro físico conterá até 127 bytes, ou seja, os códigos são blocados de 127 em 127, facilitando assim o processo de transmissão.

O registro completo ao final do processo conterá, portanto, o nome do programa, o endereço inicial, o endereço final e os códigos blocados. O modo **C** limpará a tela antes de emitir seu relatório, o qual terá a seguinte forma:



onde: **1** é o nome do arquivo; **2**, o endereço inicial; **3**, o endereço final; e **4**, o código hexa corrente. Ao fim do processo, o modo **C** retornará automaticamente ao estado **OK**.

A sintaxe do modo **D** é a seguinte:

Ela lê um arquivo gerado pelo modo **C**, desbloca os registros e aloca os códigos seqüencialmente na memória, a partir do »

Projetos & Serviços Micro Sistemas

Digitação não é mais problema

peço enviarem pelo correio o serviço

MS save (Cr$ 45.000) | MS list (Cr$ 18.000)

Programas de interesse	MS n.º, pág.	Valor

Projeto MICRO BUG

☐ Sim, desejo receber a fita MICRO BUG, com cartão de referência e manual, pelos quais pagarei Cr$ 75.000

Para isto estou enviando cheque no valor de CR$ _____ à ATI Editora.

NOME _____
ENDEREÇO _____
CIDADE _____ TEL _____
CEP _____ ESTADO _____

ATI EDITORA LTDA
Av.: Presidente Wilson, 165/1210 - CEP 20030
Rio de Janeiro/RJ - Tel.: (021) 262-6306

Leia e assine Micro Sistemas

ASSINATURA ANUAL

Se você deseja assinar MICRO SISTEMAS, preencha o cupom abaixo (ou uma cópia, caso você não queira cortar a revista).

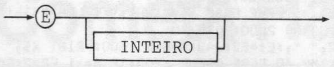
Nome _____
Profissão/cargo _____
Empresa _____
Endereço para remessa _____
Cidade _____ CEP _____ Estado _____

MICRO SISTEMAS ..... Cr$ 140.000

Preencha um cheque nominal à **ATI Editora Ltda** e envie para:
Rio de Janeiro: Av. Presidente Wilson, 165/gr. 1201
20230 - Centro - RJ - Tel. (021) 262-6306
São Paulo: Rua Oliveira Dias, 153
Jardim Paulista - 01433 - São Paulo (SP) - Tel: (011) 853-3800
Seu recibo será enviado pelo correio

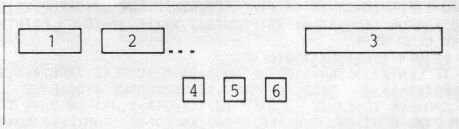
endereço inicial. Uma vez alocado o programa na memória, o modo **D** transforma o endereço inicial em novo endereço de carga, ficando o programa pronto para ser utilizado. O relatório de saída será idêntico ao gerado pelo modo anterior, também limpando a tela. Seu retorno ao modo **OK** também será automático.

E, finalmente, o modo **E** tem a seguinte sintaxe:



Ele inicializa o montador, tomando como endereço base aquele especificado (o default será o endereço de carga). O modo **E** monta os códigos após cada comando, mostrando na tela seus valores hexa. Em caso de erro, o módulo emitirá a mensagem correspondente e não incrementará o endereço.

Após a emissão do endereço em decimal e do endereço em hexa, a rotina esperará até entrarmos com o mnemônico para então processá-lo. O modo **E** gera o seguinte relatório como saída:



onde: **1** é o endereço em decimal; **2**, o endereço em hexa; **3**, o mnemônico Z-80; **4**, **5** e **6**, a instrução codificada em hexa (até 3 bytes). O modo **E** retorna ao estado **OK** como o modo **B**.

LIMITAÇÕES DO SISTEMA

A maior restrição que o usuário enfrenta é o uso de uma linguagem interpretada e, conseqüentemente, lenta: o BASIC. Certos módulos, como o **E**, que necessitam de um processamento maior, serão mais demorados. O sistema foi projetado para máquinas compatíveis com o modelo I da linha TRS-80, sem interface para disco. Assim, os módulos **C** e **D** referem-se a arquivos em fita.

O montador apresenta como limitações a não permissão do uso de pseudo-instruções; o reconhecimento somente das constantes hexa; o não processamento de símbolos; e o reconhecimento de 300 instruções do Assembler Z-80. Para contornarmos tais limitações, utilizamos o modo **B**, codificando direto em hexa as instruções não reconhecíveis ou funcionando como **DEFB** ou outras.

Um exemplo para utilização do sistema está demonstrado na figura 3. Utilize o modo **E** para codificá-la, prestando atenção nos códigos hexa. Caso algum byte esteja diferente, confira sua listagem. Após a montagem, confira os conteúdos com o modo **A** e execute a rotina. O vídeo deve ficar totalmente aceso.

```
Endereço inicial de carga = 31300

LD HL , 3C00    -  21 00 3C
LD B , 0F       -  06 0F
LD C , 40       -  0E 3F
LD , (HL) , BF  -  36 BF
INC HL          -  23
DEC C           -  0D
JP NZ , 7A4B    -  C2 4B 7A
DEC B           -  05
JP NZ , 7A49    -  C2 49 7A
RET             -  C9
```

Figura 3: Um exemplo de utilização do sistema

Helvécio Carneiro R. Netto tem 21 anos e é Matemático, formado pela UFRJ. Trabalha no desenvolvimento de software básico, na CONPART-Eletrônica, tendo ainda diversos cursos (FORTRAN, COBOL, Pascal, Assembler, ALGOL, BASIC e C) na área de informática. Ele é usuário de um TK85 e profissionalmente utiliza um IBM-PC.

Softline

O SOFTWARE PARA O SEU MICRO

Na compra de qualquer produto Softline, receba o informativo TILT com a descrição de programas, dicas, novidades e informações técnicas.

F = pode ser gravado em fita K7
J = uso obrigatório de joystick
48 = requer 48 K de RAM
64 = requer 64K de RAM

JOGOS - APPLE II

— Cr$ 80.000 —

Sabotage (F) - evite os pára-quedistas
Sargon (F) - jogo de xadrez
Draw Poker (F) - jogo de poker
Blackjack (F) - jogo de 21 (cartas)
Checkers (F) - jogo de damas
Hero (F) - resgate os mineiros perdidos
Robotron (F) - destrua os robôs inimigos
Falcons (F) - jogo tipo Phenix
Night Mission (F) - jogo de pinball
Fast Gammon (F) - jogo de gamão
Othello (F) - jogo de tabuleiro
Lode Runner - pegue os tesouros
Olympic Decation - jogos olímpicos
Space Raiders (F) - de tipo centopéia
Gorgon (F) - jogos dos Russos
The Elminator (F) - combate espacial
Night Crawler (F) - enfrente as centopéias
Space Eggs (FJ) - caçar ovos espaciais
Autobahn (F) - corrida de carros
The Asteroid Field (F) - tipo asteróides
Hard Hat Mack - perigos na construção
Choplifter (FJ) - resgate com helicóptero
Sea Dragon - pilote o submarino
Spy's Demise (F) - jogo de agente secreto
Star Blazer - combate aéreo
Viper (F) - alimente a cobra
Cavern Creatures - penetre na caverna
Móon Patrol (F) - patrulha lunar
Ms. Pac Man (F) - a namorada do Pac Man
Mario Bros (F) Mário contra o gorila
Buck Rogers - pilote sua nave em "Zoom"
Pitfall II - aventuras nas cavernas perdidas
Castle Wolfenstein - engane os nazistas
Beyond Castle Wolfenstein - mate Hitler
Aztec - decifre o enigma da pirâmide azteca
Stellar 7 - combate no espaço em 3D
Super Buny - combata as pragas
Alien Ambush (FJ) - jogo tipo Arcade
Out Post (F) - defenda a estação espacial
Defender (FJ) - defender original
Dog Fight II (F) - confronto de jatos
Star Trek - jornada nas estrelas
Jellyfish - pilote um submarino
Swashbuckler II - duelo de espadachins
Temple Apshai - aventura no templo
Flight Simulator (F) - simulador de vôo
One on One (J) - basquete com 2 jogadores
Spare Change - poupe para subir na vida

— Cr$ 170.000 —

Karateka - perfeita simulação de karatê enfrente vários adversários e liberte a princesa do Akuma. Conan - localize e destrua. Volta em várias fases e com diferentes obstáculos
Ghostbusters (J) - monte uma empresa de caça aos fantasmas e comece sua aventura
Summer Games (64) - 8 modalidades completas de atletismo, com as mais emocionantes provas olímpicas, de verão
Summer Games II (64) - 7 novas modalidades olímpicas
Winter Games (64) - 7 provas atléticas de inverno
SPY vs SPY - divertida ação de espião contra espião, baseada nos personagens da revista MAD
Droll - fantástica perseguição com várias fases. Gráficos incríveis e ação rápida
Sargon III - super jogo de xadrez
Sky Fox (J/64) - super simulador de vôo com incríveis situações de combate em vários níveis. Gráficos sofisticados
F-15 Strike Eagle (J/64) - incrível simulação de um caça a jato F-15 Eagle, com combate em vários níveis. Gráficos complexos
Pinball Constructor Set (J) - monte suas próprias mesas de fliperama
Transilvania - nesta aventura, salve a princesa Sabrina dos terríveis perigos da terra de Drácula
Archon (64) - fantástico jogo de estratégia, tipo xadrez, com animação de gráficos
S.A.M. - produz uma simulação da voz humana.
Mask of the Sun - aventura gráfica. No México, você deve achar os tesouros perdidos de uma antiga civilização
Kabul Spy - envolva-se neste thriler de espionagem tendo o Afeganistão como cenário
Dark Crystal - aventura gráfica baseada no filme "Cristal Encantado", com vários cenários
Flight Simulator II - pilote um Piper 181 totalmente equipado, através de realísticos cenários. Quando você estiver preparado enfrente um combate real
Seven Cities of Gold (J/64) - aventura gráfica. Você deve achar o Novo Mundo, retornando com ouro e glória para a Coroa
The Incrível HULK - aventura gráfica baseada na revista do HULK. Você deve dotar este incrível ser de inteligência
Capitain Goodnight (J/64) - sua missão é SALVAR O MUNDO das garras do terrível Dr. Maybe, localizando e destruindo a arma do juízo final
Rescue Raiders (J/64) - pilotando um helicóptero, comande suas tropas, durante essa incansável batalha
Bruce Lee (J) - lute karaté como o famoso Bruce Lee. Mate os ninjas e os lutadores de sumô, e complete as mais incríveis fases
Mr. Do (J) - você é um mágico que deverá proteger o seu pomar de uma terrível praga
Were is Carmen Sandiego (64) - seja um detetive que deverá prender o culpado pelo roubo. Viaje pelo mundo coletando pistas

UTILITÁRIOS/APLICATIVOS - APPLE II

— Cr$ 360.000 —

Visicalc - planilha de cálculos
Visiplot/Visitrend - gerador de gráficos
Visidex - organizador de atividades
Visifile - gerador de banco de dados
Visiterm - programa de comunicação via MODEM
Visischedule - análise financeira PERT/CPM
Apple LOGO - compilador
Apple FORTRAN (CP/M) - compilador
Apple Mechanic - utilitário gráfico
Alpha Plot - gerador de gráficos para uso comercial
GraFORTH - compilador
Locksmith 5.0/F - utilitário de cópia
Nibbles Away II/C3 - utilitário de cópia
The Graphics Megicians - animador gráfico
The Graphics Solution - editor de gráficos
Print Shop - gerador de impressos gráficos
Magic Window II - processador de textos

— Cr$ 520.000 —

Turbo PASCAL (CP/M) - compilador.
Take 1 - super animador gráfico.

JOYSTICK ANALÓGICO - APPLE II

— Cr$ 360.000 —

JOGOS - CP-300/500

— Cr$ 80.000 —

Flight Simulator (F) - simulador de vôo
Olympic Decation (F) - jogos olímpicos
Hoppy (F) - salve o sapo
Armored Patrol (F) - patrulha de tanques
Lunar Lander (F) - pouse nas crateras da lua
Alien Defense (F) - proteja-se dos invasores
Meteor Mission (F) - caçada espacial
Outhouse (F) - proteja o seu banheiro
Robot Attack (F) - destrua os robôs
Sea Dragon (F) - pilote um submarino
Missile Atack (F) - defenda sua cidade
Scarfman (F) - jogo tipo Pac Man
Cavern (F) - escape vivo da caverna
Penetrator (F) - aprofunde-se nas cavernas
Stelar Scort (F) - defenda-se da frota inimiga
Assault (F) - fuja dos ágeis inimigos
Demon Seed (F) - acabe com os enormes pássaros
Cosmic Fighter (F) - jogo tipo Invasores
Panik (F) - escape dos robôs
Pinball (F) - tipo fliperama
Crazy Painter (F) - pinte a tela, se puder!
Chicken (F) - ajude a galinha
Galaxy Invasion (F) - tipo Invasores
Sargon (F) - jogo de xadrez
Asilo (F) - aventura gráfica
Zork I - super aventura
Spook House - aventura gráfica
Toxic Dumpsite - aventura gráfica
Adventur - 4 aventuras diferentes
FucFuc - programa pornográfico
Ghosts (F) - cace os fantasmas de uma cidade
Rally - complete o trajeto de um rally

UTILITÁRIOS APLICATIVOS - CP-500

— Cr$ 450.000 —

Creator - gerador de programas BASIC
Visicalc - planilha de cálculos
Clone III - copiador rápido de discos
ACCEL 3/4 - compilador BASIC
Bascom - compilador BASIC

— Cr$ 560.000 —

NEWDOS/80 2.0 - sistema operacional
DOSPLUS 3.5 - sistema operacional
Arranger II - arquivo de diretórios
Profile III+ - gerador de banco de dados
SuperSCRIPSIT - processador de textos
ZEUS - super editor assembler
DisnDATA - super disassembler

— Cr$ 670.000 —

Producer - gerador de programas
Super Utility 3.2 - super utilitário.

JOGOS - SPECTRUM/TK 90X

— Cr$ 70.000 —

Fighter Pilot (48) - pilote um F-15 Eagle
Flight Simulation (48) - simulador de vôo
TerrorDaktil 4D (48) - ação na terra perdida
Trader (48) - comércio e ação no espaço
Styx (48) - ande no labirinto e mate a bruxa
Tutankhamun (48) - explore a tumba do faraó
Tornado (48) - pilote o avião. Cenário em 3D
Match Point (48) - fantástico tênis em 3D
Cavelon (48) - uma aventura no castelo
AD Astra (48) - salve os soldados. Ação em 3D
Galaxy Attack (48) - várias fases no espaço
Chess (48) - jogo de xadrez com vários níveis
Android 2 (48) mate os robôs. Cenário em 3D
Delta Wing (48) - pilote um caça F-16 em 3D
Manic Miner (48) - recolha os objetos da mina
Potty Painter (48) - preencha os quadrados
Pool (48) - fantástico jogo de sinuca
Pi'infere (48) - recolha as peças do computador
Piromania (48) - apague os incêndios
The Pyramid (48) - aventura na pirâmide
3D Tank Duel (48) - super batalha de tanques
3D Starstrike (48) - guerra nas estrelas
Son of Blagger (48) - recolha os objetos
Psytron (48) - defenda sua base lunar, em 3D
Wheelie (48) - pilote uma moto no subterrâneo
Fall Guy (48) - baseado no filme DURO NA QUEDA
Booga-Boo (48) - ajude o sapo salvando a mocinha
Arcventure (48) - mate o Conde Drácula
Huckback (48) - ajude o Corcunda de Notredame
Harrier Attack (48) - combate nas Malvinas
SPY v SPY (48) - guerra dos espiões, em 3D
Knight Lore (48) - aventura no castelo, em 3D
Fred (48) - desvende os perigos da pirâmide
Fantastic Voyage (48) - entre no corpo humano
Airwolf (48) - baseado no seriado ÁGUIA DE FOGO
1994 (48) - uma missão no futuro
River Raid (48) - percorra de avião um rio perigoso
Blade Alley (48) - desafio espacial em 3D
Maziacs (48) - cace um tesouro no labirinto
3D Ant Attack (48) - salve a moça das formigas
Androids (48) - aventura no labirinto
Chese Cake (48) - salve os humanos com sua nave
Scubadive (48) - recolha as pérolas do mar
Penetrator (48) - destrua o reator
Jetset Willy (48) - recolha os objetos
Full Throttle (48) - corrida de motos, em 3D
Chequered Flag (48) - mantenha a POLE POSITION
Worse at Sea (48) - mantenha o navio flutuando
Trashman (48) - ajude a manter a cidade limpa
Aquaplane (48) - uma lancha num mar de perigos
Jump Hallager (48) - pule os obstáculos
Night Gunner (48) pilote um bombardeio em 3D
Ghostbusters (48) - caça aos fantasmas
Blind Alley (48) - baseado no filme TRON
Beach Head (48) - invada uma praia inimiga protegida
Chuckie Eggs (48) - recolha os ovos em várias fases
Atic Atac 3D (48) - percorra um castelo assombrado
Mugsy (48) - controle o sindicato do crime
Sabre Wulf (48) - aventura gráfica na floresta
Autonomia (48) - saga de um montador de carros
Kokotoni Wilf (48) - perigos em várias épocas
Tirnanog (48) - aventura gráfica na idade média
Kosmic Kanga (48) - ajude um canguru voador
Pégasus (48) - destrua os alienígenas no labirinto
Special Delivery (48) - uma aventura com o Papái Noel
Lazy Jones (48) - enfrente vários videogames
Wimpy (48) - uma aventura tipo Burguertime
Decathlon 1 (48) - 5 emocionantes provas olímpicas
Decathlon 2 (48) - mais 5 provas olímpicas
Strip Poker (48) - dispa sua incrível oponente
Esquimo Eddie (48) - uma aventura na antártica
Kong (48) - ajude Mario a salvar a moça do gorila
Turti Frutti (48) - incrível caça níqueis sonoro
Pinball (48) - simulação de um pinball
Freez (48) - destrua o inimigo com os cubos de gelo
New Poker (48) - clássico jogo de poker
Horace skiing (48) - esquie montanha abaixo
Hungry Horace - divertido Pac-Man
Horace & Spiders - enfrente as aranhas
PSSST - salve sua planta das pragas
Cookie - ajude o cozinheiro contra o ZÉ DO LIXO
Tranz Am - pegue os troféus nos EUA
JetPac - construa os foguetes
Spectres - um jogo diferente tipo come-come
Escape - saia do labirinto sem ser morto
Lab 3D - será que você sai dessa?
Ground Attack - jogo tipo Penetrator
Meteors - destrua os meteoros e naves
Combate Zone - batalha de tanques em 3D
Astro Blaster - defenda o império do inimigo
Arcadia - sobreviva na travessia fatal
Space Intruder - jogo tipo Invaders
Planetoids - aventura espacial com combate
Seiddab Attack - combata os invasores da cidade
High Moon - duelo no velho oeste
Invasion Force - destrua a nave mãe
Sir Lancelot - pegue os objetos no castelo
Molar Maul - defenda seus dentes das cáries
Galaxians - destrua as abelhas assassinas
Monster In Hell - cuidado com o inferno
Space Raiders - tipo invasores do espaço

UTILITÁRIOS/APLICATIVOS - SPECTRUM/TK 90X

— Cr$ 80.000 —

VU 3D - utilitário gráfico. Telas em 3D
VU Calc - planilha de cálculos
VU File - arquivo de dados para uso geral
Ultra Violet - super disassembler
MCODER II - compilador BASIC

— Cr$ 100.000 —

SoftCALC - planilha de cálculos com GRÁFICOS
PASCAL 1.6 - compilador
SpecFORTH - compilador
ASSEMBLER - super editor assembler
Tasword II - processador de texto com 64 colunas
The KEY - super copiador de programas
Paint Plus - editor de telas gráficas
Screen Machine - utilitário gráfico

JOGOS - HOT BIT/EXPERT (MSX)

— Cr$ 80.000 —

Flight Simulator - simulador de vôo num Boeing 737
Polar Star - guerra cósmica em POLAR IV, 3D
DONPAN - divertido jogo de ação
Jumping Rabbit - ajude o coelho a subir o edifício
Project A - jogo com incríveis lutas de Kung Fu
Haunted House - escale o prédio assombrado
Trick Boy - fantástico jogo de fliperama
Cosmo Traveler - defenda-se dos inimigos espaciais
3D BOMBERMAN - elimine o monstro
Dizy Ball - destrua as pedras rolantes
Thezeus - salve a princesa presa no labirinto
Pitfall II - aventuras nas cavernas perdidas
Hiper Sports 1-3 - sensacionais competições
Hiper Sports 2 - mais 3 competições
HERO - salve os mineiros presos na mina
Moon Patrol - patrulhe a superfície da Lua
Le Mans - fantástica corrida de carros
Been Raider - defenda-se do ataque alienígena
River Raid - percorra um rio sem ser destruído
Super Cobra - defenda seu helicóptero do inimigo
Keystone Kapers - prenda o ladrão
Frogger - ajude o sapo a salvar a sapinha
Pyramid - desvende os mistérios da pirâmide
Dog Fighter - derrube os aviões inimigos
Shadow of the Bear - emocionante aventura gráfica
Binary Land - ajude os namorados a se encontrarem
Norseman - defenda seu reinõ contra os dragões
Hunch Back - salve a princesa aprisionada na torre
Hot Shoe - desligue o reator nuclear
Decatlon - várias incríveis provas olímpicas
Antartic Adventure - explore a Antártica
Spook e Ladders - escape dos terríveis monstros

Desejo receber os seguintes programas pelos quais estou enviando um cheque nominal à ATI Editora Ltda., Av. Pres. Wilson, 165 - Grupo 1210 - Centro, CEP 20030 - Rio de Janeiro - RJ. Despesas de correio já incluídas.

Computador: ( ) CP-300/CP-500 ( ) APPLE II ( ) SPECTRUM/TK 90X. Com:... K RAM. Desejo os programas em: ( ) FITA ( ) DISKETTE.

NOME: _____ ITEM(s)_____ PREÇO Cr$ _____

END.: _____ CIDADE:_____ _____ _____

UF:_____ CEP:_____ TEL.:_____ TOTAL Cr$ _____

Softline - Av. Graça Aranha, 145, Slj. 01 - Centro - Rio de Janeiro - RJ - CEP 20030 - Tel.: (021) 262-6968

Apesar de pouco conhecida, a instrução CPIR é uma das mais poderosas do Z-80. Saiba como utilizá-la em seu ZX81

A instrução CPIR nos jogos

José Armando Leite Fernandes

CPIR é uma das inúmeras instruções do Z-80. Neste artigo vou dar algumas dicas de como aproveitá-la para uso em jogos, facilitando a programação em Assembler. Vamos, inicialmente, lembrar seu significado. Quando o Z-80 encontra esta instrução, ele realiza os seguintes passos:

1– CP (compara) o acumulador (registrador A) com o conteúdo do endereço apontado pelo par HL;
2– I (incrementa) o par HL de 1;
3– R (repete) o ciclo até que o par BC seja 0 (BC é decrementado de 1 a cada ciclo) ou o resultado da comparação seja verdadeiro (Bit 0 do registrador F setado).

Vamos ver como esta instrução nos permitirá maniupular bytes do arquivo de imagem com facilidade, sem nos preocuparmos em reservar espaços na RAM para arquivar endereços. Para fixar a idéia, será mostrado (e comentado) na íntegra um pequeno jogo que serve ao propósito.

Suponha que vamos desenvolver um jogo em Assembler. Como todo bom programa, ele deve ser planejado, para que saibamos o que mandar o Z-80 fazer! A parte de impressão inicial da tela e mensagens deixamos para o BASIC, por ser uma linguagem mais adequada para este fim. Antes de digitar a parte em BASIC, carregue no micro um montador Assembler (de preferência), o Micro Bug ou o Monitor Assembler, publicado em **MS 23**. Qualquer um destes utilitários servirá para introduzir a parte em Assembler.

Após isto, então, digite esta listagem BASIC:

```
 1 REM 200 CARACTERES
10 REM JALF SOFTWARE-1985
20 FOR N=0 TO 31
30 PRINT AT 5,N;CHR$ 128
35 PRINT AT 16,N;CHR$ 128
40 NEXT N
50 PRINT AT 15,16;"A"
60 RAND
70 FOR N=1 TO 13
80 PRINT AT N,31*RND;CHR$ 19
90 NEXT N
100 POKE 16507,0
110 LET N=USR 16523
120 PRINT AT 10,2;"TIROS: ";PEEK 16507
130 IF INKEY$="" THEN GOTO 130
140 PRINT AT 10,2;"          "
150 GOTO 70
```

Não execute ainda, pois não existe a rotina em linguagem de máquina chamada na linha 110. Crie a linha 1 com 200 caracteres quaisquer, o que é mais do que suficiente.

Inclua, enquanto não armazenamos a rotina em linguagem de máquina, a linha abaixo:

```
165 STOP
```

Ela permitirá apresentar a tela sem que o micro "saia do ar". Após isto, dê **RUN**. Observe então a tela do nosso jogo. Vemos um A na parte inferior da tela. Este será o canhão, que é fixo. Para atirar – caráter : –, bastará pressionar qualquer tecla. Os aviões – caráter < – serão posicionados aleatoriamente entre as barras pretas. Quem faz isto são as linhas 70, 80 e 90. O endereço 16507 será o contador de tiros dados. O objetivo é destruir todos os aviões com o menor número de tiros. Os aviões movem-se continuamente para a esquerda, só podendo ter um tiro de cada vez na tela.

Após as regras do jogo terem sido determinadas e o BASIC ter cumprido sua tarefa, vamos ver o que o Z-80 deverá fazer (isso porque é desnecessário explicar tudo, pois ao lado de cada instrução existe um comentário):

1– Verificar se ainda existem aviões na tela; se existirem, movimentá-los para a esquerda; senão, retornar ao BASIC;
2– Verificar se existe tiro na tela; se existir, movimentá-lo para cima e retornar ao passo 1; senão,
3– Verificar o acionamento de alguma tecla. Imprimir caráter de tiro acima do canhão, caso ocorra o acionamento;
4– Voltar ao passo 1.

Estas são as principais partes. Todas possibilitam mostrar a utilidade do CPIR para localizar e verificar a existência de um determinado byte no vídeo.

Observe esta pequena rotina Assembler:

```
SUB LD HL,(16396)   ; Carrega HL com DFILE
    LD BC,768       ; Carrega BC com total de bytes
                      do arquivo de imagem
    CPIR            ; Executa procura
    RET             ; Retorna
```

Esta é a sub-rotina SUB. Vamos estudar seu funcionamento. O dado de entrada deverá ser o código do caráter a ser *procurado* no arquivo de imagem. Para isso, carregue-o no registrador A(acumulador). O retorno virá no registrador F(flag), mais especificamente no bit zero. Caso o referido byte tenha sido encontrado, o bit zero estará setado, senão, ressetado. Além dis-

so, a sub-rotina retorna à posição do byte procurado (caso exista) no par HL. Não se esqueça, porém, que CPIR incrementa HL antes de comparar BC com zero, logo a posição na realidade é apontada por HL-1.

Vamos continuar com a listagem Assembler e verificar o que foi dito acima:

```
DEL LD BC,2550    ; Carrega BC com atraso
VOL DEC BC        ; Decrementa BC de 1
    LD A,B        ; Carrega acumulador com B
    OR C          ; Verifica se BC e' zero
    JR NZ,VOL     ; Retorna se falso
                  ;
    LD A,19       ; Dado de SUB
    CALL SUB      ; Chama SUB
    RET NZ        ; Retorna ao BASIC se BIT ZERO
                  ; ressetado
```

As cinco primeiras instruções servem para tornar o programa mais lento, face à grande velocidade do Assembler. Modificando o valor carregado em BC, se obtém maiores ou menores atrasos. Logo em seguida, verifica-se a presença de aviões na tela. Caso não exista mais nenhum, o programa retorna ao BASIC para imprimir o número de tiros dados; do contrário, continua para movimentá-los:

```
      LD HL,(16396) ; Carrega HL com inicio do arquivo
                    ; de imagem
      LD BC,768     ; Carrega BC com total de bytes
                    ; do arquivo de imagem
INI   LD A,19       ; Dado de SUB
      CPIR          ; Procura
      JR NZ,TIR     ; Se nao existir mais avioes va'
                    ; para TIR
      PUSH HL       ; Salva HL para depois cont. CPIR
      DEC HL        ; Aponta HL para o byte procurado
      LD (HL),0     ; Apaga aviao
      DEC HL        ; Aponta HL para nova pos. do aviao
      LD A,(HL)     ; Carrega A com seu conteudo
      CP 118        ; Verifica se e' NEW LINE
      JR NZ,NL      ; Se nao for va' para NL
      LD DE,32      ; Carrega DE com 32

      ADD HL,DE     ; Reajusta HL
      LD (HL),19    ; Imprime aviao
      POP HL        ; Recupera HL para
      JR INI        ; continuar CPIR
NL    CP 0          ; Verifica se e' espaco
      JR NZ,VA      ; Se nao for va' para VA
      LD (HL),19    ; Imprime aviao
      POP HL        ; Recupera HL para
      JR INI        ; continuar CPIR
VA    LD (HL),0     ; Apaga a posicao apontada por HL
                    ; que so' pode conter :
      POP HL        ; Recupera HL para
      JR INI        ; continuar CPIR
```

Observe, principalmente, o início desta rotina. Todos os demais pontos retornam a ele, porque é ali que o Z-80 *procura* por um novo avião para movimentá-lo. Repare que não importa onde esteja, se existir, CPIR encontrará. Isto é muito bom, pois podemos colocar quantos aviões quisermos e em qualquer posição através do BASIC, já que a rotina movimenta todos os aviões presentes na tela. Fica muito mais fácil verificar a presença e determinar a posição dos aviões. Se, ao invés disso, guardássemos os endereços das posições dos aviões na RAM, perderíamos espaço de memória, e, principalmente, um número grande de aviões necessitaria de rotinas maiores para movimentá-los. Experimente fazer isto e veja o cansaço e dor de cabeça que dá. Muitas vezes desistimos e simplificamos o problema. Perdemos com isso a idéia inicial.

Mas não parou aí. Vamos utilizar este recurso para verificar a presença do tiro na tela (lembre-se que só pode ter um de cada vez) e movimentá-lo:

```
TIR LD A,14       ; Carrega A com dado de SUB
    CALL SUB      ; Chama SUB
    JR NZ,TEC     ; Se nao existir tiro na tela va' para TEC
    DEC HL        ; Corrige HL
    LD (HL),0     ; Apaga posicao apontada por HL
    LD DE,33      ; Carrega DE com 33
    AND A         ; Zera CARRY FLAG
    SBC HL,DE     ; Aponta HL para nova pos. de tiro
    LD A,(HL)     ; Carrega A com seu conteudo
    CP 0          ; Verifica se e' vazio
    JR NZ,CH      ; Se nao for va' para CH
    LD (HL),14    ; Imprime tiro
    JP DEL        ; Volta ao inicio
CH  CP 128        ; Verifica se chegou na barra preta
    JP Z,DEL      ; Se chegou volta ao inicio
                  ; Se nao so' pode ser aviao

    LD (HL),0     ; Retira-o da tela
    JP DEL        ; Volta ao inicio
```

Listagem do jogo

```
16514  2A 0C 40 01 00 03 ED B1 C9 01
16524  F6 09 0B 78 B1 20 FB 3E 13 CD
16534  82 40 C0 2A 0C 40 01 00 03 3E
16544  13 ED B1 20 21 E5 2B 36 00 2B
16554  7E FE 76 20 09 11 20 00 19 36
16564  13 E1 18 E7 FE 00 20 05 36 13
16574  E1 18 F5 36 00 E1 18 F9 3E 0E
16584  CD 82 40 20 1D 2B 36 00 11 21
16594  00 A7 ED 52 7E FE 00 20 05 36
16604  0E C3 8B 40 FE 80 CA 8B 40 36
16614  00 C3 8B 40 2A 25 40 2C CA 8B
16624  40 3E 26 CD 82 40 11 22 00 A7
16634  ED 52 36 0E 3A 7B 40 3C 32 7B
16644  40 C3 8B 40
```

Só falta agora verificarmos o acionamento ou não de alguma tecla. Se houver, vamos imprimir um tiro acima do canhão. Note que para determinarmos a posição do canhão usamos mais uma vez CPIR, não precisando calcular nem guardar a sua posição no arquivo de imagem.

```
TEC LD HL,(16421) ; Carrega HL com variavel do sistema ULTK
    INC L         ; Verifica se alguma tecla e' acionada
    JP Z,DEL      ; Se nao, volta ao inicio
    LD A,38       ; Carrega A com dado de SUB
    CALL SUB      ; Chama SUB
    LD DE,34      ; Carrega DE com 34
    AND A         ; Zera o CARRY FLAG
    SBC HL,DE     ; Aponta HL para nova pos.
    LD (HL),14    ; Imprime tiro
    LD A,(16507)  ; Carrega A com o contador de tiros
    INC A         ; Incrementa contador
    LD (16507),A  ; Salva novo valor
    JP DEL        ; Volta ao inicio
```

Após ter acompanhado todo este raciocínio, pode-se chegar às seguintes conclusões:

1º) Não se gastou RAM guardando a posição dos aviões, canhão, e tiro. Cada uma exigiria dois bytes;
2º) Poderia ter qualquer número de aviões em qualquer posição, pois CPIR acharia cada um deles;
3º) Pode-se facilmente determinar a existência ou não do tiro e dos aviões;
4º) Não foi preciso calcular a posição inicial dos aviões e do canhão.

Obviamente que no caso do canhão seria mais eficiente calcular e guardar na RAM a sua posição, pois ele é fixo e sua posição inicial é sempre a mesma. Porém, como o Assembler é muito rápido, isto não se torna sensível, e o resultado final não fica comprometido. Pessoalmente, prefiro usar CPIR do que calcular endereços.

Digite todas as rotinas aqui apresentadas, caso o leitor possua um Montador Assembler; se não, entre com a listagem dos códigos hexadecimais. Retire a linha 105. Grave o programa. Rode-o e observe os resultados. Mude a listagem BASIC de forma a colocar o canhão em outra posição na linha em que se encontra. Diminua ou aumente o número de aviões. Coloque-os em outras posições. Comprove o alívio que CPIR pode lhe oferecer na programação Assembler. Bom proveito!

José Armando Leite Fernandes é Aspirante do quarto ano da Escola Naval, cursando Habilitação em Sistemas de Armas.

Se você tem um micro da linha TRS-80 e deseja proteger seus programas contra a ação dos curiosos e *piratas*, aqui estão algumas técnicas simples que, certamente, poderão ajudá-lo.

Léo ataca os piratas

Léo Luiz Ferraz Netto

Um dos maiores problemas atuais é a proteção de programas contra a ação dos *piratas*. Mesmo não havendo uma segurança inviolável e total, é possível, pelo menos, complicar a vida destes "espertinhos". Com o objetivo de auxiliar o leitor, descreverei algumas técnicas utilizadas e o modo de implementá-las em seus programas.

BLOQUEIO DO BREAK

De fato, uma vez que a tecla **BREAK** é reconhecida na operação de varredura do teclado (*scan keyboard*), podemos nos aproveitar disso. O vetor de desvio fica armazenado nos endereços 16396, 16397 e 16398, com valores default (modo cassete) 201, 0 e 0, respectivamente. Para anular a ação do **BREAK**, basta iniciar o programa (a ser gravado ou não) com **POKE 16396,23** e, para reativar, colocar no modo de execução, no final do programa: **6000 POKE 16396, 201**. Lembre-se, o bloqueio do **BREAK** impede a listagem!

PROGRAMA COM SENHA

A chamada (**GOSUB**) dessa sub-rotina, logo no início do programa principal, determina que o programa propriamente dito só seja executado após a entrada de uma senha, que nesse exemplo é LEO – o que não deve ser feito na prática, pois a primeira coisa que o *pirata* vai tentar é o nome do autor, suas iniciais, etc. Ao terminar a execução, o sistema entra em loop, e somente sai dele, voltando ao comando do teclado, se uma determinada tecla código (que nesse exemplo é "!") for acionada.

```
5 GOSUB 1000
10 CLS
   .
   .
   .  CORPO PRINCIPAL DO
   .  PROGRAMA
   .
   .
999 GOTO 2000
1000 POKE 16396,23
1001 INPUT"ENTRE COM A SENHA DE
COMANDO DE EXECUCAO";SS$
1002 IF SS$()"LEO",K=K+1:IF K(5,
PRINT"TENTATIVA#: ";K;PRINT" VOC
E ESTA AUTORIZADO?":GOTO 1001
1003 IF SS$()"LEO" AND K=5,CLS:P
RINT@460,"PROGRAMA BLOQUEADO A P
ESSOAL NAO AUTORIZADO";@908,"CHA
ME PROGRAMADOR RESPONSAVEL":GOTO
 2000
1004 CLS:PRINT@460,"SENHA CORRET
A...BOM PROVEITO!";
1005 FOR T=1 TO 1600:NEXT:RETURN
2000 CC$=INKEY$:IF CC$="",2000
2001 IF CC$="!",PRINT"RETORNO AO
 TECLADO...BLOQUEIO DESATIVADO":
POKE 16396,201:END ELSE 2000
```

COMPLICANDO O CÓDIGO

No programa anterior, a rotina de retorno ao comando do teclado (2000-2001) poderá ser bastante sofisticada, utilizando, por exemplo, as técnicas de codificação da matriz do teclado. Desse modo, a rotina acima pode ser substituída por:

```
2000 AA=PEEK(14337):IF AA=6,POKE
 16396,201 ELSE 6000
```

Neste caso, o loop só se abrirá – e você conseguirá assumir o comando do programa pelo teclado – se no endereço 14337 houver o conteúdo 6, e isto só é conseguido de um modo: apertando simultaneamente as teclas **A** e **B**. Para justificar isso, veja o capítulo **Matriz do Teclado**, em seu manual, e observe quantas composições você poderá fazer. No exemplo dado, trabalhando apenas no endereço 14337, o leitor poderá trocar o 6 por 1, 2, 3, 4, 5, 8, 9, 10, 16, 17, 18, 32, 33, 34, 64, 65, 128, 129. Quanto ao BREAK será reativado com as teclas: @, A, B, C, D, E, F, G, @ + A, @ + B, @ + C, @ + D, @ + E, @ + F, @ + G, A + C, A + D, A + E, etc. sucessivamente.

BLOQUEIO DO LIST

O primeiro número de linha do seu programa, seja ele qual for (0, 1, 10, 1000, etc.), fica armazenado em dois bytes (LSB e MSB) nos endereços 17387 e 17388. Desse modo, se 10 é o seu primeiro número de linha, em 17387 teremos o conteúdo 10 e em 17388 teremos o conteúdo 0, pois 10 + 0 * 256 = 10. Outro exemplo: se 500 é o seu primeiro número de linha, o conteúdo do endereço 17387 será 244 e o do endereço 17388 será 1, pois 244 + 1 * 256 = 500. Alterando estes conteúdos, você afetará apenas a listagem, sem interferir na execução do programa. Ao terminar seu programa, dê no modo de execução o seguinte comando:

```
>POKE 17387,254:POKE 17388,255
```

Desse modo, ao se solicitar a listagem (**LIST**), nada ocorrerá, pois se o primeiro número de linha do programa é o último número permitido, não há programa e, portanto, nada há para listar!

Para recuperar a listagem, basta recolocar naqueles endereços seus conteúdos originais, ou seja, se o primeiro número da linha original é 10, coloque:

```
>POKE 17387,10:POKE 17388,0
```

LINHA CONTRA PIRATA

Já que o *pirata* quer ver sua listagem, para alterá-la levemente, colocando o nome dele no programa, vamos dar uma linha especial para ele. Inicie seus programas sempre assim:

```
0 PRINT"SAI PIRATA!!!":CLS
```

Isso em nada vai alterar seu programa e nem será lido, tal a rapidez de execução e limpeza de tela. Ao terminar seu programa normal, e antes de gravá-lo em fita, coloque:

```
>POKE 16836,67
```

Com isso, seu programa ou qualquer cópia dele terá sua listagem bloqueada, exceto a primeira linha, que é a do *pirata*!

Para recuperar sua listagem integral, faça:

```
>POKE 16836,201
```

ZERANDO NÚMEROS DAS LINHAS

Permita ao *pirata* listar o seu programa, porém com todos os números de linha zerados. Para complicar ainda mais o trabalho dele, habitue-se a utilizar em seus programas **números de linha** bem exdrúxulos. Que tal digitar um programa começando com:

```
>AUTO 13,27
```

Veja só que lista linda: 13, 40, 67, 94, 121, 148, etc. Agora que seu programa está todo pronto, antes de gravá-lo, vamos zerar todos os números de linha. Se o programa não for muito extenso, isso pode ser feito "na raça", consultando a área do programa, verificando os endereços que contêm os números de linha e zerando-os (as chamadas de GOTO e GOSUB podem ser deixadas). Vamos fazer isso "automaticamente". Eis um exemplo e a sub-rotina:

```
13 PRINT"SAI PIRATA!!!":CLS
40 PRINT"CORPO DO PROGRAMA"
67 '
94 '
121 '
148 '
  .
  .
  . Fim do seu
  . Programa
  .
  .
6000 E=17385
6001 X=PEEK(E):Y=PEEK(E+1):A=X+2
56*Y
6002 IF X=0 AND Y=0,END
6003 POKE E+2,0:POKE E+3,0
6004 E=A:GOTO 6001
```

A seguir, faça:

```
>RUN
?UL Erro na 6004
>DELETE 6002-6004
```

E agora:

```
>LIST
0 PRINT"SAI PIRATA!!!":CLS
0 PRINT"CORPO DO PROGRAMA"
0'
0'
0'
0'
  .
  .
  .
0'FIM DO SEU PROGRAMA
0 E=17385
0 X=PEEK(E):Y=PEEK(E+1):A=X+256*
Y
Ready
>
```

Este programa não pode ser editado para alterações, pois está protegido (?)!. Ele poderá ser copiado, mas os desvios internos deixarão o pirata louco.

Lembre-se: *Pirata* não estuda, não lê revistas técnicas a fundo e só procura "chupar" coisas que são do entendimento superficial dele!

Nota: Se você quer mesmo ser "malvado com os *piratas*", troque a linha 6003 (antes de dar **RUN**) pela nova:

```
6003 POKE E+2,RND(250):POKE E+3,
RND(250)
```

Barrabás! que malvadeza! (com o leitor, pois isso vai embaralhar todo o seu programa e você vai perdê-lo, Oh! Oh! Oh!).

As dicas aqui apresentadas foram testadas num microcomputador TRS-80, modelo III, utilizando o BASIC residente. Algumas delas, contudo, rodam em BASIC disco.

Léo Luiz Ferraz Netto é Professor de Física do Colégio e Curso Objetivo. Ele é autodidata e aprendeu a programar através de livros e revistas.

NOTA 10!

Antes de inscrever-se em qualquer curso de informática, conte até 10. Conheça a escola, os métodos e equipamentos. Confira se o que você vai aprender bate com o que você realmente precisa. O SENAC-INFORMÁTICA é a escola mais completa de São Paulo. Tem cursos práticos para quem quer aprender a operar, programar, conhecer linguagens, aplicativos, ou simplesmente, jogar com o micro. O SENAC-INFORMÁTICA está oferecendo 10 novos cursos nota 10 da linha Sinclair. Escolha o seu e venha tirar a prova dos 9. Você vai concordar em número, gênero e grau!

1. Operação TK90X e Introdução ao Basic Sinclair
2. Programação Basic TK90X
3. Assembler Z-80
4. Assembler Z-80 Avançado
5. Criação e Programação de Jogos no TK90X
6. Criação e Programação de Aplicativos no TK 90X
7. FORTH no TK90X
8. PROLOG no TK 90X
9. Aplicações Profissionais do TK90X
10. Logo para Crianças

INSCRIÇÕES ABERTAS.
VAGAS LIMITADAS!

Informações e Inscrições:

Rua Dr. Vila Nova, 228
Tel.: (011) 255-0066

SPP

senac
informática

Estratégia

Jorge A. C. Bettencourt Soares

Você é o piloto de uma aeronave em missão de bombardeio e está chegando muito perto do alvo. Mas agora, antes de alcançar seu objetivo, você está sobrevoando (rumo norte) uma faixa do território inimigo, intensamente defendida por baterias antiaéreas e mísseis terra-ar.

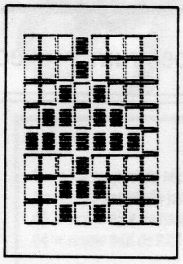
Na tela do radar, à sua frente, você tem as informações necessárias à tomada de decisões estratégicas. A faixa clara vertical, no centro do vídeo, é o corredor aéreo que você está sobrevoando, subdividido por linhas pontilhadas que demarcam quatro rotas identificadas pelos dígitos 1, 2, 9 e 0.

A posição atual do seu bombardeiro é assinalada por um cursor negro, piscando no centro da tela, precisamente na rota 2. Seu avião está sendo perseguido por um míssil inimigo, localizado por um cursor na forma de um asterisco pulsante.

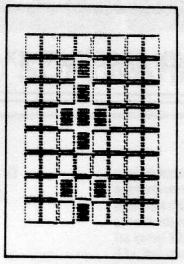
Se você conseguir ultrapassar o trecho final do corredor aéreo, ficará fora do alcance do míssil e poderá, então, lançar sua carga de bombas sobre o objetivo, retornando à base para nova missão. Para isto, a melhor estratégia a seguir consiste em avançar em ziguezague, desorientando o míssil, pulando sucessivamente de uma rota para outra do modo mais aleatório possível. Lembre-se que o sistema de radar inimigo que guia o míssil estará registrando e analisando todos os seus movimentos e tentará adivinhar, uma a uma, as rotas que você escolhe, direcionando o míssil para lá. Toda a vez que você ludibriar o míssil, desviando para uma rota diferente da dele, sua nave avançará uma posição.

```
  10 REM ***** ESTRATEGIA ******
  20 REM *        COM          *
  30 REM *    INTELIGENCIA     *
  40 REM *     ARTIFICIAL      *
  50 REM ************************
  60 REM JORGE A. C. B. SOARES
  70 FAST
  80 RAND
  90 POKE 16418,0
 100 DIM A(16,4)
 110 DIM D$(24)
 120 DIM E$(100)
 130 LET MC=0
 140 LET MI=0
 150 LET CA=15
 160 LET CM=15
 170 LET A$=CHR$ 0
 180 FOR N=1 TO 24
 190 LET D$(N)=A$
 200 NEXT N
 210 LET B$=D$(1 TO 3)
 220 FOR N=1 TO 100
 230 LET K=RND
 240 IF K<.7 THEN LET E$(N)=CHR$
 129
 250 IF K>=.7 THEN LET E$(N)=CHR
$ 128
 260 NEXT N
 270 FOR N=1 TO 16
 280 FOR K=1 TO 4
 290 LET A(N,K)=RND
 300 NEXT K
 310 NEXT N
 320 GOSUB 1000
 330 FOR N=1 TO 2
 340 LET I=INT (4*RND)
 350 GOSUB 3000
 360 NEXT N
 370 GOSUB 2000
 380 GOSUB 3000
 390 GOSUB 4000
 400 GOTO 370
1000 REM TELA DO RADAR
1010 FAST
1020 LET D=9
1030 LET LA=12
1040 LET LM=22
1050 FOR N=0 TO 23
1060 PRINT "                : : : :
                "
1070 NEXT N
1080 PRINT AT 4,3;"    ";AT 4
,5-LEN STR$ MC;MC;AT 4,10-LEN ST
R$ MI;MI;AT 5,3;"MC   MI"
1090 PRINT AT 14,3;"   N   ";AT
15,3;"   *   ";AT 16,3;" O++*E "
;AT 17,3;"   *   ";AT 18,3;"   S
   "
1100 FOR N=1 TO 5
1110 GOSUB 1160
1120 NEXT N
1130 PRINT AT 23,13;"1:2:9:0"
1140 SLOW
1150 RETURN
1160 LET X=1+INT (92*RND)
1170 PRINT AT D,22;E$(X TO X+8)
1180 LET D=D+1
1190 IF D=14 THEN LET D=9
1200 RETURN
2000 REM ESTRATEGIA DO INIMIGO
2010 LET X=1+4*VAL B$(2)+VAL B$(
3)
2020 LET K=0
2030 FOR Y=1 TO 4
2040 IF A(X,Y)>K THEN GOSUB 2070
2050 NEXT Y
2060 RETURN
2070 LET K=A(X,Y)
2080 LET I=Y-1
2090 RETURN
3000 REM ESTRATEGIA DO PILOTO
3010 LET DF=PEEK 16396+256*PEEK
16397
3020 LET E=DF+33*LA+CA+1
3030 LET F=DF+33*LM+CM+1
3040 LET B=0
3050 POKE F,23
3060 LET B=B+1
3070 POKE F,0
3080 POKE E,128
3090 GOSUB 1160
3100 POKE E,0
3110 IF B>30 THEN GOTO 3270
3120 IF INKEY$="" THEN GOTO 3050
3130 LET C=CODE INKEY$
3140 IF C<>28 AND C<>29 AND C<>3
0 AND C<>37 THEN GOTO 3040
3150 IF C=29 THEN LET A=0
3160 IF C=30 THEN LET A=1
3170 IF C=37 THEN LET A=2
3180 IF C=28 THEN LET A=3
3190 LET CM=13+2*I
3200 LET CA=13+2*A
3210 LET B$=B$(2 TO 3)+STR$ A
3220 IF CA=CM THEN GOTO 3280
3230 LET LA=LA-1
3240 IF SGN LA=-1 THEN GOSUB 500
0
3250 IF B>30 OR B$(1)=A$ THEN GO
TO 380
3260 RETURN
3270 LET CM=CA
3280 LET LM=LM-3
3290 IF LM<=LA THEN GOSUB 5140
3300 GOTO 3250
4000 REM CONTAGEM DE EVENTOS
4010 LET X=1+4*VAL B$(1)+VAL B$(
2)
4020 LET A(X,A+1)=A(X,A+1)+1
4030 RETURN
5000 REM MENSAGENS DE ALERTA
5010 CLS
5020 LET MC=MC+1
5030 LET C$="ALVO A 2 MILHAS"
5040 GOSUB 5240
5050 LET C$="LANCANDO CARGA DE B
OMBAS"
5060 GOSUB 5240
5070 LET C$="MISSAO CUMPRIDA"
5080 GOSUB 5240
5090 LET C$="RETORNANDO A BASE"
5100 GOSUB 5240
5110 CLS
5120 GOSUB 1000
5130 RETURN
5140 CLS
5150 LET MI=MI+1
5160 LET CM=13+2*INT (4*RND)
5170 IF CM=CA THEN GOTO 5160
5180 LET C$="EMERGENCIA"
5190 GOSUB 5240
5200 LET C$="BOMBARDEIRO ATINGID
O"
5210 GOSUB 5240
5220 LET C$="MISSAO INTERROMPIDA
"
5230 GOTO 5080
5240 FOR N=1 TO 4
5250 PRINT AT 10,0+(32-LEN C$)/2
;C$
5260 FOR K=1 TO 3
5270 NEXT K
5280 PRINT AT 10,4;D$
5290 NEXT N
5300 FOR N=1 TO 20
5310 NEXT N
5320 RETURN
```

Listagem 1



Letra A do UDG (avião).



Letra B do UDG (míssil).

Mas se você fracassar nesse intento, sua rota coincidirá com a do míssil que, então, avançará três posições de uma só vez, enquanto você não progride.

Para desviar seu avião para determinada rota, acione a tecla correspondente ao número da rota. Por exemplo, se você está na rota 2 e quer desviar para a rota 0, à direita, basta teclar 0. Se você demorar muito a tomar uma decisão, o míssil entrará na sua rota e avançará automaticamente três posições a cada 30 pulsos de cursor.

Um mostrador na parte superior esquerda do vídeo contabiliza o saldo de missões cumpridas (MC) e interrompidas (MI).

INTELIGÊNCIA ARTIFICIAL

Na média das partidas, o piloto consegue terminar as duas ou três primeiras missões com certa facilidade, mas daí para a frente vai ficando cada vez mais difícil ganhar do computador. A cada batalha aérea travada, a máquina aprende a jogar ainda melhor e vai aperfeiçoando progressivamente seu sistema de jogo até um ponto em que vence praticamente todas as batalhas.

Isto acontece graças à rotina de IA que decide qual a melhor rota para desviar o míssil, com base na análise dos 64 registros estatísticos do array A(16,4). Observe os procedimentos de pesquisa de probabilidades na sub-rotina 2000 e a atualização permanente das estatísticas na sub-rotina 4000.

Em algumas batalhas, a eficácia dos mísseis é tão espetacular que quase podemos jurar que o programa lhes favorece. Em termos matemáticos, as chances são iguais para ambas as partes. Vejamos: considerando que existem quatro rotas alternativas, a probabilidade do míssil adivinhar corretamente para qual o avião vai desviar é de uma em quatro. Portanto, em cada quatro missões, o míssil vai acertar uma vez, avançando três posições, enquanto que o piloto vai conseguir se esquivar três vezes, avançando três posições. Viu o empate?

Mas, na verdade, é o piloto que tem a seu favor um expressivo *handicap*, já que seu avião inicia cada batalha aérea dez posições na frente do míssil. É justamente esta vantagem que lhe dá a possibilidade de ganhar, na média, oito em

```
 170 LET A$=CHR$ 32
 240 IF K<.7 THEN LET E$(N)=CHR$
 142
 250 IF K>=.7 THEN LET E$(N)=CHR
$ 143
1060 IF N<22 THEN PRINT INVERSE
1;"                    "; INVERSE 0;"
  "; INVERSE 1;"
1065 IF N>21 THEN PRINT #0;AT N-
22,0; INVERSE 1;"                    ";
 INVERSE 0;" "; : : "; INVERSE 1;
"
1130 PRINT #0;AT 1,13;"1:2,9:0"
3050 IF LM<22 THEN PRINT AT LM,C
M;"+"
3055 IF LM>21 THEN PRINT #0;AT L
M-22,CM;"+"
3070 IF LM<22 THEN PRINT AT LM,C
M;" "
3075 IF LM>21 THEN PRINT #0;AT L
M-22,CM;" "
3080 IF LA<22 THEN PRINT AT LA,C
A;"▲"
3085 IF LA>21 THEN PRINT #0;AT L
A-22,CA;"▲"
3100 IF LA<22 THEN PRINT AT LA,C
A;" "
3105 IF LA>21 THEN PRINT #0;AT L
A-22,CA;" "
3140 IF C<>48 AND C<>49 AND C<>5
0 AND C<>57 THEN GOTO 3040
3150 LET A=(C=50)+2*(C=57)+3*(C=
48)
5240 DIM X$(96): LET X$((64-LEN
C$)/2 TO )=C$
5250 FOR N=1 TO (64-LEN C$)/2+LE
N C$+1
5260 PRINT AT 10,0;X$(N TO N+31)
5270 SOUND ,05,-10
5280 NEXT N
5290 RETURN
9998 FOR F=0 TO 21: DIM L$(80):
FOR G=0 TO 31: LET L$(G+4)=SCREE
N$ (F,G): NEXT G: LPRINT L$;: NE
XT F: STOP
9999 INPUT "APAGAR A PARTIR DE.
.";A: FOR F=A TO 21: PRINT AT F
,0,;: NEXT F: GOTO 9998
```

Listagem 2

cada dez batalhas, caso adote estratégias adequadas. Se você quiser comprovar estas afirmações, inclua as seguintes instruções no programa:

3005 LET A = INT (4 * RND)
3105 GOTO 3190

Agora dê um RUN que zera os registros estatísticos de A(16,4) e estará iniciado um ciclo automático de missões de bombardeio. Você vai observar uma série de missões onde o computador joga contra ele mesmo, utilizando a função RND para tomar as decisões do piloto, e comprovará aquilo que previmos: se tomar decisões randômicas, o piloto cumpre 80% das missões.

O ser humano não consegue tomar decisões absolutamente casuais mesmo que se esforce ao máximo para isso, e são justamente os padrões repetitivos, ocultos na seqüência de jogadas do piloto, que a IA capitaliza, favorecendo os mísseis em combate.

Pode-se alterar o grau de dificuldade do jogo, manipulando a distância inicial que separa o avião do míssil, definida pelas variáveis **LA** e **LM**, nas instruções 1030 e 1040. Mantendo fixo o valor de **LM**, se você diminuir ligeiramente o valor de **LA**, digamos para 11 ou 10, estará facilitando o jogo para o piloto, e vice-versa.

PARA O TK90X*

Se você já possui um micro compatível com o ZX Spectrum, poderá também utilizar este programa. Para isso, elimine da listagem 1 as linhas 70; 90; 1010; 1140; 3010; 3020; 3030; 3160; 3170; 3180; 5300; 5310 e 5320 e entre com as linhas da listagem 2 (para o TK90X).

Com as modificações citadas, o programa roda nos micros ZX Sepctrum, TS 2068 e TK90X. Se você fizer modificações nos comandos PRINT e acrescentar efeitos sonoros em algumas passagens do programa, poderão ainda ser melhor aproveitadas as capacidades gráficas e sonoras destes equipamentos; use sua imaginação. Na listagem 2, nas linhas 3050; 3055; 3080 e 3085, as letras entre aspas correspondem às letras A e B dos caracteres do UDG (a letra A representa o avião, e a B, o míssil, que no programa original são representados por um caráter invertido e um asterisco; faça você mesmo a redefinição destes dois caracteres).

* As modificações efetuadas no programa, para que o mesmo rode em compatíveis com o ZX Spectrum, foram feitas por Divino C. R. Leitão.

Jorge A. C. Bettencourt Soares é Engenheiro Agrônomo e trabalha na Coordenadoria de Assistência Técnica Integral (CATI), da Secretaria de Agricultura e Abastecimento, em Bauru, SP.

ALBAMAR ELETRÔNICA LTDA.

• DIGITAPE C'5	– Cr$ 11.200
• DIGITAPE C'10	– Cr$ 12.800
• DIGITAPE C'15	– Cr$ 13.600
• DIGITAPE C'20	– Cr$ 14.400
• DIGITAPE C'30	– Cr$ 16.800
• DIGITAPE C'46	– Cr$ 17.600
• DIGITAPE C'60	– Cr$ 19.000
• BASF C'60	– Cr$ 22.400
• SCOTH C'60	– Cr$ 20.800
• FITA DE LIMPEZA DE CABEÇOTE	– Cr$ 14.000
• FITA VÍDEO K-7 T 120	– Cr$ 170.000
• MICROFONE MK2 LESON	– Cr$ 290.000
• TWEETER LESON 100W	– Cr$ 55.000
• MÓDULO P/ K-7	– Cr$ 12.000

PEDIDOS POR CARTA ACOMPANHADOS DE CHEQUE NOMINAL A **ALBAMAR ELETRÔNICA LTDA.** RUA CONDE DE LEOPOLDINA, 270-A – SÃO CRISTÓVÃO – RJ – CEP: 20.930 – TEL.: (021) 580-6729.
ACRESCENTAR 10% DO VALOR DA COMPRA PARA DESPESAS DE CORREIOS.

Analisador de header

Paolo Fabrizio Pugno

Certamente, quem já prestou um pouco de atenção ao sistema de operação com cassete do ZX Spectrum ou TK90X percebeu que sua grande característica é a variedade de comandos disponíveis, aliada a uma grande segurança na transferência de dados do micro para a fita e vice-versa. Porém, é muito normal que queiramos fazer um back-up de segurança de um determinado programa (nunca se sabe...), e é aí que geralmente começam os problemas.

Enquanto o programa a ser copiado for só em BASIC, não teremos maiores dificuldades, pois basta dar um BREAK e, logo em seguida, um banalíssimo SAVE "nome" LINE xx.

Mas é muito freqüente que os programas contenham uma rotina em linguagem de máquina, a qual é carregada logo que o programa em BASIC começa a rodar através de um LOAD " " CODE. Porém, neste caso, fica muito difícil, senão impossível, com procedimentos normais, sabermos qual o comprimento em bytes desta rotina e qual é seu endereço inicial. Isto porque o LOAD "" CODE, sem parâmetro nenhum, assume como *default* o endereço original do bloco de bytes no momento em que ele foi salvo. De alguma forma, porém, o micro deve ser informado destes parâmetros. E é justamente aqui que a *porca torce o rabo...* Aonde vamos buscar estas informações?

O HEADER

Antes de carregar qualquer informação vinda do cassete, o Spectrum tem que saber se trata-se de um programa BASIC, um bloco de bytes, uma matriz numérica ou alfanumérica, para, só então, estabelecer o modo de colocação na memória. O que fornece estas informações é o header ou, simplesmente, o cabeçalho – aquela curta descarga de sons que antecede cada bloco salvo, seja ele um programa, matriz etc. Ele é composto por 17 bytes assim subdivididos:

O primeiro byte informa sobre o tipo de bloco que se segue, podendo assumir quatro valores distintos: 00 (indica programas em BASIC); 01 (indica matriz numérica); 02 (indica matriz alfanumérica) e 03 (indica bloco de bytes).

Os dez bytes seguintes contêm o nome do bloco em questão. Se o nome tiver menos de dez caracteres, o espaço restante será preenchido com códigos ASCII 32 (space). Isto também explica aquela estranha obsessão do micro em não aceitar nomes com mais de dez caracteres. O 12º e 13º bytes contêm o número total de bytes que serão carregados na memória.

O 14º e 15º indicam a linha de programa para a qual o micro deve saltar logo após o carregamento do programa em BASIC, se este foi salvo com SAVE "nome" LINE xx. Caso o bloco seja somente se bytes (primeiro byte do header igual a três), eles indicam o endereço de início do bloco de bytes.

O 16º e 17º devem ser considerados somente se o bloco se referir a um programa em BASIC. Neste caso, eles contêm o número de bytes relativos ao programa sem variáveis. Logicamente, se antes de efetuarmos o SAVE dermos um CLEAR (procedimento recomendado), o valor destes bytes será igual ao indicado pelo 12º e 13º bytes.

Para podermos analisar corretamente o header e, assim, obtermos estas informações, preparei um pequeno programa, o qual contém uma pequena rotina em linguagem de máquina. Esta rotina predispõe o micro a guardar, em um local da memória estabelecido por nós, um certo número de bytes também determinado por nós. Observe como ele funciona e veja sua estrutura principal:

```
LD DE, 17
LD IX, 23372
LD A, 0
SCF
CALL 1366
```

No par de registradores DE colocamos o número de bytes que iremos carregar; IX contém o endereço a partir do qual vamos carregar estes bytes; o acumulador deve conter 0 e devemos setar o *carry flag* (caso contrário, ao invés de um LOAD executaremos um VERIFY). Logo após, chamamos a rotina da ROM no endereço (decimal) 1366, e a brincadeira estará feita. Naturalmente, podemos carregar quantos bytes quisermos, somente alterando o valor de DE, e a partir de qualquer endereço alterando IX. No nosso caso, resolvi colocar a rotina na área reservada no buffer da impressora pelos seguintes motivos:

- O mais certo seria alocá-la no topo da memória, mas isso demandaria duas versões do mesmo programa, uma para quem tem 16 e outra para quem tem 48 Kb de memória. Melhor não complicar.
- Poderia colocá-la numa linha 1 REM, como se fazia nos micros compatíveis com o ZX81, porém no Sepctrum isso não é recomendável, pelo seguinte motivo: microdrives. Eles ainda não foram lançados, mas têm toda a *pinta* de que serão, e muita gente já possui os originais da Sinclair, por meios oficiais ou não. Acontece que a interface que controla os microdrives desloca a área BASIC em, pelo menos, 58 bytes à frente, para criar novas variáveis de sistema, e isso prejudicaria o funcionamento normal do programa.

Optando pela solução mais simples, resolvi colocar, seja o programa, sejam os dados, no buffer da impressora. Este buffer começa no endereço 23296 e é longo (256 bytes), assim, temos espaço de sobra.

Quem já tiver alguma experiência com programação vai perceber que o programa é mais extenso do que o que foi descrito: não há nenhum problema, apenas coloquei uns "embelezamentos" que podem ser facilmente eliminados. Mas, mesmo assim, não creio que o programa seja tão longo a ponto de provo-

car desânimo só em pensar em digitá-lo (muito pelo contrário).

DESCRIÇÃO DO PROGRAMA

Linhas 10 e 20: contêm os códigos que serão pokeados no buffer da impressora, contendo a rotina em linguagem de máquina;
Linha 30: loop que lê um dado e efetua os POKEs;
Linha 40: faz a chamada para a rotina em 1/m (linguagem de máquina) e, conforme o tipo de header analisado, manda para a sub-rotina BASIC correspondente;
Linhas de 100 a 130: imprimem as informações relativas a um programa em BASIC;
Linha 200: imprime as informações relativas a uma matriz numérica;
Linha 300: imprime as informações relativas a uma matriz alfanumérica;
Linha 400: imprime as informações relativas a um bloco de bytes;
Linha 500: sub-rotina utilizada por todas as outras, imprime o nome contido no header e o número total de bytes do bloco, bem como calcula os outros dois parâmetros.

VARIÁVEIS UTILIZADAS

n1 – contém o número total de bytes do bloco;

```
  1 REM ********************
  2 REM *   ANALISADOR     *
        *   DE  HEADER     *
  3 REM ********************
  4 REM
  5 REM
  10 DATA 62,2,205,1,22,33,35,91
,6,41,126,215,35,16,251,17,17,0,
221,33,76,91,62,0,55,205,86,5,33
,76,91,6,0,78,201
  20 DATA 22,10,0,17,3,16,6,18,1
,32,79,75,32,76,73,71,85,69,32,7
9,32,71,82,65,86,65,68,79,82,32,
69,32,65,71,85,65,82,68,69,46,32
  25 REM
  30 RESTORE : FOR f=0 TO 75: RE
AD a: POKE (23296+f),a: NEXT f
  40 BORDER 7: CLS : GOTO 100+10
0*USR 23296
  100 CLS : BORDER 1: PRINT PAPER
 4;"PROGRAMA EM BASIC"'': GOSUB
500: PRINT PAPER 5;"No. de bytes
 de programa: "; PAPER 7;n3''
  110 IF n2<=9999 THEN PRINT PAPE
R 5;"LINE: "; PAPER 7;n2''
  120 IF n1<>n3 THEN PRINT PAPER
3; INK 9;"Bytes de variaveis: ";
 PAPER 7;n1-n3: STOP
  130 PRINT PAPER 3; INK 9;"O PRO
GRAMA NAO CONTEM VARIAVEIS.": ST
OP
  200 CLS : BORDER 2: PRINT PAPER
 4;"MATRIZ NUMERICA"'': GOSUB 50
0
  210 STOP
  300 CLS : BORDER 3: PRINT PAPER
 4;"MATRIZ ALFANUMERICA"'': GOSU
B 500
  310 STOP
  400 CLS : BORDER 4: PRINT PAPER
 4;"BLOCO DE BYTES"'': GOSUB 500
  410 PRINT PAPER 5;"Endereco ini
cial: "; PAPER 7;n2: STOP
  500 PRINT PAPER 5;"Nome: "; FO
R f=1 TO 10: PRINT PAPER 7;CHR$
PEEK (23372+f);: NEXT f: LET n1=
PEEK 23383+256*PEEK 23384: PRINT
'' PAPER 5;"No. total de bytes:
 "; PAPER 7;n1'': LET n2=PEEK 23
385+256*PEEK 23386: LET n3=PEEK
23387+256*PEEK 23388: RETURN
1000 SAVE CHR$ 18+CHR$ 1+"HEADER
" LINE 1
```

Analisador de header

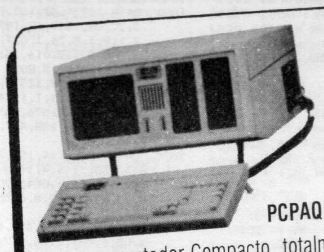
n2 – contém o LINE (caso o bloco seja um programa BASIC e este tenha um autostart) ou o endereço inicial do bloco (caso o header se refira somente a um bloco de bytes);
n3 – contém o número de bytes de programa BASIC (sempre que o bloco for um programa BASIC).

Para finalizar, aí vão duas observações importantes: 1) quem tem a impressora Alphacom 32 pode ter alguns problemas com este programa, pois ele dificulta o acesso ao buffer. Neste caso, é recomendável desconectá-la. Quem possui uma ZX Printer ou TK Printer não precisa se preocupar; 2) pode acontecer, em alguns casos, do header conter um caráter de controle, e o mesmo, ao ser impresso, causar erro K (cor inválida) ou algo semelhante. Caso isto aconteça, simplesmente digite CONT, e o programa prosseguirá mostrando os dados normalmente.

Paolo Fabrizio Pugno cursa atualmente o primeiro ano da Faculdade de Engenharia Industrial (SP) e possui um ZX Spectrum, com o qual desenvolve programas.

COISA DE MACRO
OS MICROS DA COMPUTERWARE!

Para quem pensa grande, a Computerware tem sempre os melhores preços do mercado, com estoques suficientes para uma entrega imediata e condições de pagamento a sua escolha.
Além de descontos especiais, a Computerware oferece assistência técnica com profissionais altamente especializados, o melhor atendimento, contrato de manutenção e uma infinidade de vantagens, que fazem a nossa macro diferença.



PCPAQ.
Microcomputador Compacto, totalmente compatível com o IBM PC/XT. Duas reduzidas placas de circuito, o PCPAQ reúne no mesmo gabinete: CPU, memória inicial de 256 K, monitor de vídeo 9'', 2 drives de 360 K, além de 2 slots adicionais para expansões e comunicações de dados. É incrível a versatilidade do PCPAQ, com um peso total de 13 kg, pode ser levado para onde quiser, protegido por uma bolsa almofadada. ... Consulte-nos.

UNITRON Ap II
Computador pessoal. Microprocessador 6502, 48 K de memória RAM e 12 K de memória ROM, teclado com maiúsculas e minúsculas, 8 conectores para expansões.

UNITRON T.I.
Além de reunir as características do APII, possui o Teclado Inteligente Unitron, que, diretamente, como em máquinas de escrever, fornece maiúsculas / minúsculas e acentuação da língua portuguesa. Apresenta a característica especial de permitir a programação de cada tecla com comandos ou funções definidos pelo usuário.
Na Computerware, encontram-se também a disposição, todas as interfases / placas e periféricos do Unitron. ... Consulte-nos.

★ **ASSISTÊNCIA TÉCNICA** ★
(021) 262-1886
• Técnicos especializados na fábrica
• Atendimento imediato
• Contratos de manutenção

VISITE NOSSO SHOW-ROOM
Rio - INFOSHOPPING
Rua do Catete, 311

COPYRIGHT

COMPUTER WARE Informática Ltda.

Rio - Av. Almirante Barroso, 91 - 11º andar - Tel.: (021) 240-7294
Rio - INFOSHOPPING - R. do Catete, 311 - (Show Room)
São Paulo - Rua Jesuíno Arruda, 797/Grupo 22 - Tel.: (011) 881-7446
Curitiba - Av. João Gualberto, 1375 - 1º andar - Tel.: (041) 253-5433

Cometa Halley

Antônio Fernando B. de Almeida Prado

Uma das maiores atrações dos últimos tempos é, sem dúvida alguma, a passagem do cometa Halley. Com o programa publicado a seguir (escrito para microcomputadores compatíveis com o Apple II, mas facilmente adaptável para outras famílias), você pode encontrar dados sobre o cometa Halley (e mais 14 cometas periódicos) em qualquer data (passada ou futura) e estar preparado para observá-lo.

Antes de entrarmos em detalhe sobre o programa, acho que seria interessante darmos algumas informações sobre os cometas. Eles são pequenos blocos feitos de gelo e poeira (gelo aqui não significa apenas gelo de água, mas de qualquer substância como amônia etc.). Seria apropriado chamá-lo de pedra de gelo suja. Essas pedras foram expulsas das proximidades do Sol na época da formação do Sistema Solar e agora mantêm-se a uma distância equivalente à metade do caminho até a estrela mais próxima. Esse fato é de grande importância, pois os cometas guardam em seus corpos as características básicas da época da formação dos planetas. Olhar para eles agora equivale a voltar no tempo e olhar para a Terra (ou outro planeta) na época da sua formação. Esse fato motiva a comunidade científica do mundo inteiro a apontar seus instrumentos, para eles quando se aproximam de nós. Agora, pela primeira vez na História da Humanidade, lançamos sondas ao seu encontro, para melhor estudá-lo.

O aparecimento de um cometa no céu é resultado de um jogo de influências gravitacionais na região habitada por eles. Esse jogo acaba lançando o cometa para a região próxima ao Sol e, a

```
]PR#0
]LIST

10  REM  COMETAS
20  REM  ANTONIO FERNANDO BERTACH
     INI
30  HOME
40  PRINT "ESCOLHA O COMETA:": PRINT

50  DIM C$(15)
60  FOR I = 1 TO 15
70  READ C$(I)
80  NEXT I
90  DATA  ENCKE,TEMPLE 2,HANEDA-C
    AMPOS,SCHWASSMANN-WACHMANN 2
    ,BORRELLY,WHIPPLE,OTERMA,SCH
    AUMASSE,COMAS SOLA,SCHWASSMA
    NN-WACHMANN 1,NEUJMIN 1,CROM
    MELIN,OLBERS,PONS-BROOKS,HAL
    LEY
100  FOR I = 1 TO 15
110  PRINT I;"- ";C$(I)
120  NEXT I
130  INPUT A
140  GOSUB 1000
150  HOME : PRINT "DATA (DD,MM,AA
     AA):"
160  INPUT DIA,MS,YR
170  GOSUB 2000
180 Y = DA - PE
190 MC = 360 * Y / TP
200  IF MC > 0 THEN 220
210 MC = MC + 360: GOTO 200
220  IF MC < 360 THEN 240
230 MC = MC - 360: GOTO 220
240 MC = MC * 3.1416 / 180
250  GOSUB 3000
260 V =  SQR ((1 + EM) / (1 - EM)
    ) *  TAN (E / 2)
270 V =  ATN (V) * 2:V = V * 180 /
    3.1416
280 LC = V + WB
290 RC = AM * (1 - EM * EM) / (1 +
    EM *  COS (V * 3.1416 / 180)
    )
300 PSI =  SIN ((LC - OM) * 3.141
    6 / 180) *  SIN (IO * 3.1416
     / 180)
310 PSI =  ATN (PSI /  SQR (1 - P
    SI * PSI))
320 Y =  SIN ((LC - OM) * 3.1416 /
    180) *  COS (IO * 3.1416 / 1
    80)
330 X =  COS ((LC - OM) * 3.1416 /
    180)
340 AU =  ATN (Y / X)
350  IF X < 0 THEN AU = AU + 3.14
    16
360  IF Y < 0 AND X > 0 THEN AU =
     AU + 2 * 3.1416
370 AU = AU * 180 / 3.1416
380 LL = OM + AU
390  IF LL > 0 THEN 410
400 LL = LL + 360: GOTO 390
410  IF LL < 360 THEN 430
420 LL = LL - 360: GOTO 410
430 RL = RC *  COS (PSI)
440  REM  CALCULOS PARA A TERRA
450  GOSUB 5000
460 NE = 360 * D / TE / 365.2422
470  IF NE > 0 THEN 490
480 NE = NE + 360: GOTO 470
490  IF NE < 360 THEN 510
500 NE = NE - 360: GOTO 490
510 ME = NE + EP - WT
520 L = NE + 360 * ET *  SIN (ME *
     3.1416 / 180) / 3.1416 + EP
530  IF L < 360 THEN 550
540 L = L - 360: GOTO 530
550  IF L > 0 THEN 570
560 L = L + 360: GOTO 550
570 VE = L - WT
580 RT = (1 - ET * ET) / (1 + ET *
     COS (VE * 3.1416 / 180))
590  IF RL > RT THEN 630
600 AJ = RL *  SIN ((L - LL) * 3.
    1416 / 180) / (RT - RL *  COS
    ((L - LL) * 3.1416 / 180))
610 LAM = 180 + L +  ATN (AJ) * 1
    80 / 3.1416
620  GOTO 650
630 AJ = RT *  SIN ((LL - L) * 3.
    1416 / 180) / (RL - RT *  COS
    ((LL - L) * 3.1416 / 180))
640 LAM = LL +  ATN (AJ) * 180 /
    3.1416
650  IF LAM > 0 THEN 670
660 LAM = LAM + 360: GOTO 650
670  IF LAM < 360 THEN 690
680 LAM = LAM - 360: GOTO 670
690 BET = RL *  TAN (PSI) *  SIN
    ((LAM - LL) * 3.1416 / 180) /
    RT /  SIN ((LL - L) * 3.1416
     / 180)
700 BET =  ATN (BET)
710 LAM = LAM * 3.1416 / 180
720  GOSUB 4000
730 PP =  SQR (RC * RC + RL * RL -
    2 * RC * RL *  COS (PSI))
740 RO = PP /  SIN (BET)
750 RO =  ABS (RO)
810  HOME
815  PR# 1
820  PRINT "COMETA: ";C$(A)
830  PRINT
840  PRINT "DATA: ";DIA;"/";MS;"/
    ";YR
850  PRINT
860  PRINT "DECLINACAO: ";
870  IF GA = 0 AND DO < 0 THEN  PRINT
    "-";
880  PRINT GA;" GRAUS ";MIN;" MIN
    "
890  PRINT
900  PRINT "ASCENCAO RETA: ";AH;"
     HORAS ";K5;" MIN"
910  PRINT
920  PRINT "DISTANCIA DA TERRA EM
     U.A.: ";RO
930  PRINT
940  PRINT "DISTANCIA AO SOL EM U
    .A.: ";RC
942  PRINT : PRINT : PRINT
945  PR# 0
950  END
1000  REM  SUBROTINA QUE FORNECE
    OS DADOS
1010  DIM D(105)
1020  FOR I = 1 TO 105
1030  READ D(I)
1040  NEXT I
1050  DATA  1974.32,160.1,334.2,3
    .3,2.209,.847,12,1972.87,310
    .2,119.3,5.26,3.024,.549,12.
    5,1978.77,12.016,131.7,5.37,
    3.066,.64152,5.805
1060  DATA  1974.7,123.3,126,6.51
    ,3.489,.386,3.7,1974.36,67.8
    ,75.1,6.76,3.576,.632,30.2,1
    970.77,18.2,188.4,7.47,3.821
    ,.351,10.2
1070  DATA  1958.44,150,155.1,7.8
    8,3.958,.144,4,1960.29,138.1
    ,86.2,8.18,4.054,.705,12,196
    9.83,102.9,62.8,8.55,4.182,.
    577,13.4
1080  DATA  1974.12,334.1,319.6,1
    5.03,6.087,.105,9.7,1966.94,
    334,347.2,17.93,6.858,.775,1
    5,1956.82,86.4,250.4,27.89,9
    .173,.919,28.9
1090  DATA  1956.46,150,85.4,69.4
    7,16.843,.93,44.6,1954.39,94
    .2,255.2,70.98,17.2,.955,74.
    2,1986.112,170.011,58.154,76
    .0081,17.9435,.9673,162.2384

1100 PE = D(7 * (A - 1) + 1)
1110 WB = D(7 * (A - 1) + 2)
1120 OM = D(7 * (A - 1) + 3)
1130 TP = D(7 * (A - 1) + 4)
1140 AM = D(7 * (A - 1) + 5)
1150 EM = D(7 * (A - 1) + 6)
1160 IO = D(7 * (A - 1) + 7)
1170  READ TE,EP,WT,ET
1180  DATA  1.00004,98.83354,102.
    596403,.016718
1190  RETURN
2000  REM  SUBROTINA QUE CALCULA
    A DATA EM ANOS DECIMAIS
2010  DIM AU(12)
2020  FOR I = 1 TO 12
2030  READ AU(I)
2040  NEXT I
2050  DATA  0,31,59,90,120,151,18
    1,212,243,273,304,334
2060 DA = DIA + AU(MS)
2070  IF (YR / 4 -  INT (YR / 4) =
    0) AND (MS > 2) THEN DA = DA
     + 1
2080 DA = DA / 365.25 + YR
2090  RETURN
3000  REM  SUBROTINA QUE RESOLVE
    A EQUACAO DE KEPLER
3010 E = MC
3020 D9 = E - EM *  SIN (E) - MC
3030  IF  ABS (D9) < .000001 THEN
    3070
3040 D8 = D9 / (1 - EM *  COS (E)
    )
3050 E = E - D8
3060  GOTO 3020
3070  RETURN
4000  REM  SUBROTINA QUE TRANSFOR
    MA COORDENADAS ECLIPTICAS EM
     EQUATORIAIS
4010 E1 = 23.441884 * 3.1416 / 18
    0
4020 ANG =  SIN (BET) *  COS (E1)
     +  COS (BET) *  SIN (E1) *
     SIN (LAM)
4030 D1 =  ATN (ANG /  SQR (1 - A
    NG * ANG))
4040 D1 = D1 * 180 / 3.1416
4050 D0 = D1:D1 =  ABS (D1)
4060 YS =  SIN (LAM) *  COS (E1) -
     TAN (BET) *  SIN (E1)
4070 AL =  ATN (YS /  COS (LAM)) *
    180 / 3.1416
4080  IF  COS (LAM) < 0 THEN AL =
    AL + 180
4090  IF YS < 0 AND  COS (LAM) >
    0 THEN AL = AL + 360
4100 A1 = AL / 15
4110 FR = A1 -  INT (A1)
4120 AH =  INT (A1)
4130 K5 =  INT (60 * FR)
4140 F1 = D1 -  INT (D1)
4150 GA =  INT (D1)
4160 GA =  SGN (D0) * GA
4170 MIN =  INT (60 * F1)
4180  RETURN
5000  REM  SUBROTINA QUE CALCULA
    A DIFERENCA DE DIAS
5010 D7 = DS:M7 = MS:Y7 = YR
5020  IF M7 <  > 1 AND M7 <  > 2 THEN
    5040
5030 Y7 = Y7 - 1:M7 = M7 + 12
5040 A7 =  INT (Y7 / 100)
5050 B7 = 2 +  INT (A7 / 4) - A7
5060  IF (YR < 1582) OR (YR = 158
    2 AND MS < 10) OR (YR = 1582
     AND MS = 10 AND DIA < 15) THEN
    B = 0
5070 C7 =  INT (365.25 * Y7)
5080 D8 =  INT (30.6001 * (M7 + 1
    ))
5090 JD = B7 + C7 + D8 + DIA + 17
    20994.5
5100 D = JD - 2444238.5
5110  RETURN
```

Cometa Halley

partir daí, ele fica preso à sua volta, tornando-se um cometa periódico.

Um cometa periódico executa uma alongada elipse em torno do Sol e, de tempos em tempos, passa pelo periélio (ponto mais próximo do Sol). Quando essa época se aproxima, o calor começa a sublimar (passa do estado sólido para o gasoso) o material do cometa. Assim ele fica a brilhar, adotando uma cauda. Dessa maneira o cometa acaba se tornando visível aqui na Terra. Sua cauda sempre aponta na direção oposta ao Sol, devido a um fluxo de partículas e radiações emanadas pelo Sol. Com o seu afastamento do periélio, o cometa perde o brilho e a cauda, sumindo de nossa vista para uma região mais afastada do Sistema Solar, tornando-se incógnito, mesmo aos instrumentos mais possantes até a sua próxima passagem.

O PROGRAMA

Cometa Halley é composto por um programa principal e cinco sub-rotinas. A parte principal (entre as linhas 10 e 950) tem a função de pedir os dados (input); executar os cálculos necessários e imprimir os resultados na tela e na impressora. Caso não se tenha uma impressora, deve-se suprimir as linhas **815 PR ≠≠ 1** e **945 PR ≠≠ 0**, para obter os resultados apenas na tecla. Os dados de entrada são: 1) Nome do cometa (Encke; Temple 2; Haneda – Campos; Schwassmann – Wachmann 2; Barrelly; Wipple; Otermo; Schaumasse; Comas Sola; Schwassmann – Wachmann 1; Neiymin 1; Crommelin; Olbers; Pons – Brooks e Halley) e 2) Datas no formato DD, MM, AAAA (DD = dia; MM = mês e AAAA = ano). Os outros dados são: 3) Declinação; 4) Ascensão Reta; 5) Distância da Terra (em unidades astronômicas) e 6) Distância do Sol (também em unidades astronômicas). Uma unidade astronômica equivale à distância da Terra ao Sol, valendo aproximadamente 149.000.000 Km.

Declinação e Ascensão Reta são as coordenadas de um astro no céu. Com elas você pode encontrar o cometa no céu, no máximo, com a ajuda de um pequeno mapa celeste, caso não se tenha alguma experiência (use ainda as estrelas como referência).

A sub-rotina entre as linhas 1000 e 1190 fornece os elementos orbitais do cometa, que são os dados sobre sua órbita necessários para localizá-lo em função do tempo.

A sub-rotina seguinte (entre as linhas 2000 e 2090) converte a data desejada como uma fração decimal do ano (por exemplo, transforma 15 de janeiro de 1986 em 1986, 0410677. A parte fracionária corresponde aos 15 dias transcorridos durante o ano de 1986).

Entre as linhas 3000 e 3070 temos a sub-rotina que resolve a equação de Kepler. Ela é a desenvolvida, ou melhor, resolvida por um processo iterativo, pois não existe solução analítica. As próximas sub-rotinas (4000 e 4180 e entre 5000 e 5110) fazem uma transformação de coordenadas celestes e o cálculo da diferença de dias entre a data desejada e uma data base (onde são tomados os elementos orbitais), respectivamente.

COMO EXECUTAR O PROGRAMA

Após digitá-lo corretamente, basta dar um **RUN** e seguir o código do cometa (conforme o menu) e, a seguir, fornecer a data no formato pedido. No item DD (dia) você pode usar um número fracionário para especificar a hora. Por exemplo, caso você deseje conhecer os dados do cometa às 20 horas do dia 7 de abril de 1986, deve-se inserir 7.8333,04, 1986, onde 8333 adicionado ao 7 correspondente às 20 horas (20/24 = 0.8333) transcorridos a partir do dia 7. Assim sendo, um número inteiro do dia corresponde a meia noite de início daquele dia.

PRECISÃO DO PROGRAMA

O cálculo da órbita de um cometa envolve muitos fatores difíceis de se conhecer. A distribuição de massa no Sistema Solar e a perda de massa do próprio cometa a cada passagem não são conhecidas com precisão infinita. Mesmo agora, com o cometa já redescoberto, existe discussão quanto à posição do cometa, horário do periélio etc. Este programa é geral e não leva em conta estes detalhes. Ele não foi feito para essa passagem, em particular, e sim para obter a posição em milênios. Portanto, não espere grande precisão. Porém, para amadores ela é mais do que suficiente.

Agora, é só apanhar seu instrumento (mesmo que seja apenas o olho) e sair procurando. Ainda existe um lugar de honra para o primeiro ser humano a ver o Halley a olho nu, nesta aparição.

Antônio Fernando B. de Almeida Prado é formado em Engenharia Química, pela Poli-USP, e em Física, pela IFUSP. Também possui cursos de FORTRAN, BASIC, Assembler e PL/M.

CP500/M,(80) uma grande decisão para homens que tomam decisões.

O CP 500/M80 agora também é compatível com o CP/M, o Sistema mais difundido no mundo. Isso significa que, somando o tradicional DOS 500 com o CP/M, você passa a dispor da maior biblioteca de programas aplicativos do mercado. E você poderá tirar vantagens de programas como Calc Star*, dBase II*, Word Star* e muitos outros, que vão agilizar tanto sua vida quanto a de sua empresa. Para você, executivo, o CP 500/M80 é uma poderosa ferramenta de auxílio a decisões gerenciais, com programas específicos em sua área de atuação, como análise de fluxo de caixa, controles de compras/custos/estoques e administração financeira. Como resultado final, sua empresa terá grandes ganhos de produtividade e eficiência. CP 500/M80, um grande passo na informática com a Qualidade CP e a Tecnologia Prológica.

Características	Sistemas Operacionais Inclusos	
Memória RAM	SO-08 64 Kbytes	DOS 500 48 Kbytes
Memória ROM	2 Kbytes	16 Kbytes
Compatibilidade	CP/M*	TRS-DOS
Vídeo	80 colunas por 24 linhas	64 ou 32 colunas por 16 linhas

Vídeo 12" fósforo verde com controle de intensidade
Teclado profissional capacitivo com numérico reduzido
Interface para impressora paralela
Opera com 1 ou 2 drives de face simples ou dupla
Acesso a toda a Rede Nacional de Telemática, através de Interface serial padrão RS-232 C (opcional)

* Marcas Registradas

CP
COMPUTADORES PESSOAIS
TECNOLOGIA PROLOGICA

Copydir

Marcel H. C. Tarrisse da Fontoura e Carlos E. de Moura Magalhães

Este utilitário é muito interessante para aqueles que possuem um microcomputador da linha TRS-Color com unidades de disco.

Nesta linha de microcomputadores, a perda de sistemas em disco ocorre com muita freqüência devido a problemas no diretório do disco. Na trilha 17, fica armazenado um conjunto de informações vital para o sistema operacional poder ler ou gravar programas e arquivos. A razão da perda de informações nesta trilha ainda não está completamente esclarecida. Parece que isso ocorre devido a um mau contato da interface de disco ou, então, em função de uma magnetização excessiva da cabeça de leitura e gravação. Fala-se, inclusive, em erros dentro do controlador FD 1793.

O fato da cabeça de leitura e gravação permanecer em repouso sobre a trilha 17, a maior parte do tempo, contribui para facilitar a danificação desta trilha tão importante. Quando ocorre uma falha deste tipo, o diretório fica danificado e não podemos acessar os dados gravados no disco. Assim, todos os programas e arquivos ficam perdidos.

O programa que vamos apresentar, a seguir, pode ser usado para eliminar este problema, pois ele faz uma cópia do diretório, permitindo ainda que o recuperemos a qualquer momento, mesmo se o disco estiver danificado.

O Copydir foi escrito em BASIC, usando alguns comandos do Extended Color BASIC. Ele copia a trilha 17 na trilha 0 (zero). Quando você executa o programa, surge na tela um menu com as seguintes alternativas:

1 – Salvar diretório
2 – Recuperar diretório
3 – Fim de processamento

1 – SALVAR DIRETÓRIO

Se você escolher esta alternativa, o programa vai tentar salvar o diretório (trilha 17) na trilha 0. Antes de fazer a cópia, ele verifica se a trilha 0 está livre. Se ela não estiver ocupada, um código especial (**CHR$ 191**) é gravado no byte 0 do setor 2 da trilha 17. Com esta gravação estaremos reservando a trilha 0 para copiar o diretório do disco. No setor 2 da trilha 17 fica armazenada a tabela de alocação de arquivos e programas (*File Allocation Table*). Você poderá obter maiores detalhes sobre a estrutura dos discos

```
10 'PROGRAMA DE PROTECAO DO DIRETORIO
20 'VERSAO 0.0 * CARLOS EDUARDO DE MOURA MAGALHAES *
40 CLEAR4000
50 GOSUB1000:PRINT@164,"1 .. SALVAR DIRETORIO":PRINT@196,"2 .. RECUPERAR DIRETOR
IO":PRINT@228,"3 .. FIM DE PROCESSAMENTO":PRINT@297,"INDIQUE A OPCAO ";
55 LINE INPUT R$ : IF R$("1" OR R$)"3" THEN 50
60 IF R$="1" THEN GOSUB 150
70 IF R$="2" THEN GOSUB 500
75 IF R$="3" THEN 90
80 GOTO 50
90 GOSUB1000:PRINT@232,"FIM DE PROGRAMA":END
110 GOSUB1000:PRINT@196,"A AREA DE COPIA JA' ESTA'       OCUPADA. NAO E' POSSIVE
L       SALVAR O DIRETORIO !":SOUND20,2:END
120 GOSUB1000:PRINT@195,"NAO E' POSSIVEL RECUPERAR       O DIRETORIO.ELE AINDA N
AO       FOI SALVO!":SOUND20,2:END
140  ROTINA DE PROTECAO DA TRILHA 0 (GRANULO 0)
150 DSKI$ 0,17,2,A$,B$
160 IF LEFT$(A$,1)=CHR$(255) OR LEFT$(A$,1)=CHR$(191) THEN 200
170 GOTO110
200 A$=CHR$(191)+RIGHT$(A$,127)
210 DSKO$ 0,17,2,A$,B$
215 'ROTINA DE LEITURA DA TRILHA 0
220 GOSUB1000:PRINT@165,"CONTEUDO ATUAL DA COPIA"
225 FOR T=1TO800:NEXTT
230 FOR SN=1 TO 9
240 DSKI$ 0,0,SN,D1$,D2$
245 PRINT@224,STRING$(128," ");STRING$(128," ");
250 PRINT@224,D1$:D2$;
260 FOR T=1TO500:NEXTT
270 NEXT SN
280 GOSUB1000:PRINT@200,"VOCE QUER COPIAR":PRINT@232,"O DIRETORIO S/N";:INPUT R$
290 IF R$="S" THEN 320
300 IF R$="N" THEN 50
310 GOTO 280
315 'ROTINA PARA COPIAR A TRILHA 17 PARA A TRILHA 0
320 VERIFY ON
330 FOR SN=2 TO 10
340 DSKI$ 0,17,SN,D1$(SN-1),D2$(SN-1)
350 NEXT SN
360 FOR SN=1 TO 9
370 DSKO$ 0,0,SN,D1$(SN),D2$(SN)
380 NEXT SN
390 VERIFY OFF
400 RETURN
500 'ROTINA PARA VERIFICAR SE A TRILHA 0 E UMA COPIA DO DIRETORIO
510 DSKI$ 0,0,1,A$,B$
520 IF LEFT$(A$,1)()CHR$(191) THEN 120
540 'ROTINA PARA RECUPERAR O DIRETORIO
550 GOSUB1000:PRINT@260,"RECUPERAR DIRETORIO S/N ";:INPUTR$
560 IF R$="N" THEN 50
```

```
570 IF R$="S" THEN 590
580 GOTO550
590 VERIFY ON
600 FOR SN=1 TO 9
610 DSKI$ 0,0,SN,D1$(SN),D2$(SN)
620 NEXT SN
630 FOR SN=2 TO 10
640 DSKO$ 0,17,SN,D1$(SN-1),D2$(SN-1)
650 NEXT SN
660 VERIFY OFF
670 RETURN
1000 CLS:PRINT@12,"COLOR 64";
1005 PRINT@43,"UTILITARIO";
1010 PRINT@69,"PROTECAO DE DIRETORIO"
1020 PRINT@96,STRING$(32,"-");
1030 RETURN
```

Copydir

lendo o capítulo 11 do manual Color DOS (do Color 64). No manual do CP 400 este assunto não é tratado com a devida importância.

2 – RECUPERAR DIRETÓRIO

Com esta alternativa, copiamos a trilha 0 para a trilha 17. Antes de fazer a gravação, o programa mostra na tela todos os dados contidos na trilha 0, de forma a permitir uma verificação visual das informações que serão transferidas. Esta alternativa também pode ser usada se desejarmos recuperar um programa ou arquivo apagado pelo comando **KILL**.

Quando você precisar recuperar um disco danificado, será necessário carregar o programa Copydir de um outro disco. A melhor forma de verificar se um diretório está com defeito é executar o comando **DIR**. Se aparecer a mensagem **I/O ERROR**, isso quer dizer que o disco não pode mais ser acessado.

3 – FIM DE PROCESSAMENTO

Esta alternativa apaga o programa Copydir da memória e libera o microcomputador para uso habitual.

Utilizando normalmente um microcomputador da linha TRS-Color com disco, devem ocorrer uma ou duas perdas de disco por mês. Estas perdas são devidas, em mais de 95% dos casos, a problemas na trilha 17 (diretório). O programa Copydir permite recuperar estes discos gastando apenas dois grânulos (de um total de 68).

Digite o programa com muito cuidado porque os comandos **DSKO$** e **DSKI$** são usados com freqüência e eles podem destruir o conteúdo do disco se codificados erradamente.

Para destruir o diretório durante os testes do utilitário Copydir, execute a seguinte rotina:

```
10 A$ = " "
20 B$ = " "
30 DSKO$ 0,17,2,A$,B$
```

Marcel H. C. Tarrisse da Fontoura é Engenheiro formado pelo IME. Trabalha atualmente na LZ Equipamentos como Gerente de Software.

Carlos E. de Moura Magalhães é estudante de Informática da PUC. Trabalha atualmente na LZ Equipamentos como Projetista de Firmware.

SOFTWARE NASAJON

FOLHA DE PAGAMENTO
40 ORTN'S

Faz a Folha de Pagamento de sua empresa, emitindo relatórios como Guia de IAPAS, Guia de FGTS, Relação de Empregados, Relação para I.R., Relação para Banco, Informe de Rendimentos, Acumulados Anuais, RAIS e Recibo de Pagamento.
A folha pode ser semanal ou mensal.
As tabelas são modificadas pelo próprio usuário.
Permite também, adiantamentos de salário, reajuste salarial, alterações de acumulados e outras funções que agilizem o processamento da Folha de Pagamento da empresa.

CONTABILIDADE
40 ORTN'S

A Contabilidade de um mês em apenas 2 horas! Este Sistema permite o cadastramento de históricos padronizados e de plano de contas com até 5 níveis.
Emite Diário, Razão, Balancete, Balanço, Demonstração de Resultados. Demonstração de Lucros e Prejuízos acumulados, Listagem por centro de custo e extrato de contas, entre outras funções.

CONTROLE DE ESTOQUE
30 ORTN'S

Controla o estoque de itens com Especificação, Estoque Mínimo, Unidade, Fornecedor, Localização e outras informações relacionadas no item como Custo Médio, Entradas e Saídas no período, etc.
Fornece Listagens Geral e Parcial dos produtos, Listagem Físico-Financeira, Listagem dos produtos abaixo do estoque mínimo, Lista de Preços e Etiquetas, entre outras.
Admite também, Reajuste de Preços, Alteração de Dados e Exclusão de Produtos.

Compatíveis com as linhas TR5-80 e Apple.

A NASAJON oferece assistência técnica total, garantia permanente e mantém à sua disposição programadores e analistas para desenvolver sistemas específicos sob encomenda. Conte com a NASAJON SISTEMAS.

nasajon sistemas

Av. Rio Branco, 45 - Grupo 1.311
Rio de Janeiro - CEP 20.090
Tels.: (021) 263-1241 e 233-0615

Empresa filiada à ASSESPRO.

Protector

Carlos Henrique Fraeb e Ricardo Holmer Hodara

O princípio do programa é simples. Baseia-se no fato de que podemos gravar um programa qualquer como se fosse um arquivo de dados em código ASCII, no qual cada registro de até 255 bytes é uma linha numerada do programa original. A partir daí, podemos chamar os registros ASCII e modificar seu final, colocando automaticamente os caracteres de controle que serão usados para realizar a proteção contra listagens.

Os caracteres colocados no final de cada registro ASCII (linha de programa convertido) serão os caracteres de retrocesso de cursor, caráter de limpeza (clear) e de avanço de página. Isso acontece após a colocação da token REM, que pode e deve ser substituída por outra token escolhida pelo usuário, de forma a personalizar o segredo, garantindo assim a exclusividade de acesso necessária a qualquer proteção.

OPERAÇÃO DO SISTEMA

Primeiramente, deve ser feita a gravação do programa chamado, em formato ASCII. O programa BASIC a ser protegido também deve ser gravado em formato ASCII, o que é fácil: basta chamar o programa para a memória e depois gravá-lo no disco com o seguinte comando direto **SAVE" nome do programa original + sufixo "/ASC"", A.** Isto indica ao micro que é para gravar o programa *tokenizado*, que está na memória, em seu equivalente em ASCII. Depois disso, teremos dois programas iguais gravados no disco, só que um deles mais curto, em formato de tokens, e outro mais longo, em formato ASCII.

A segunda etapa é simples, basta rodar o programa **PROTECTOR**, que já deve ter sido digitado e gravado no disco. Após a execução desse programa, teremos o mesmo protegido, já gravado no disco, no formato ASCII e com o nome **"Nome do programa Original + sufixo "/PRT""**. Para poupar bytes no disco, o usuário deve ainda chamar o programa protegido e, logo em seguida, gravá-lo com o mesmo nome no disco. Dessa forma, ele será convertido da forma ASCII para a forma *tokenizada*, que é mais curta.

Carlos Henrique Fraeb e Ricardo Holmer Hodara são estudantes de Engenharia Eletrônica da Universidade Federal do Rio Grande do Sul e da PUC-RS. Há cerca de dois anos são diretores da Comércio de Sistemas e Equipamentos Ltda.

```
19000 REM NOTA : NA ALTURA DA EXECUCAO DO PROGRAMA PROTECTOR, O
USUARIO JA DEVERA TER GRAVADO EM FORMATO ASCII O PROGRAMA ORIGIN
AL A SER PROTEGIDO (ATRAVES DO COMANDO SAVE"NOME DO PROGRAMA",A)
; QUE DEVERA SER O PROTEGIDO.
20000 CMD"DIR":CLEAR1000:REM MOSTRA O DIRETORIO DO DISCO, CASO V
OCE ESTEJA USANDO DOS500 SUBSTITUA CMD"DIR" POR CMD"D:0".RESERVA
 1000 BYTES PARA USO DE STRINGS
20010 A$="":REM"+CHR$(34)+CHR$(12)+CHR$(28)+CHR$(31):REM COMPOSIC
AO DA VARIAVEL A$, QUE E' COMPOSTA PELA TOKEN REM SEGUIDA POR AS
PA (CHR$(34)), PELOS CARACTERES DE CONTROLE DE AVANCO DE PAGINA,
 RETROCESSO DO CURSOR E CLEAR RESPECTIVAMENTE.
20020 INPUT"NOME DO PROGRAMA A PROTEGER ";B$:REM COLOCAR O NOME
DO PROGRAMA A SER PROTEGIDO, (COPIA DO ORIGINAL JA EM FORMATO AS
CII), PARA DEPOIS SER TRANSFORMADO.
20025 Z=INSTR(B$,"/"):IFZ()0THENBP$=LEFT$(B$,Z-1)+"/PRT"ELSEBP$=
B$+"/PRT":REM COLOCA A POSICAO DA BARRA DO NOME DO PROGRAMA A SE
R PROTEGIDO, CASO EXISTA, NA VARIAVEL Z. SE HOUVER ESTABELECE BP
$ A PARTIR DO NOME ANTERIOR A BARRA E ACRESCENTA O SUFIXO"/PRT",
20026 REM CASO CONTRARIO SIMPLESMENTE ACRESCENTA O SUFIXO /PRT
20030 OPEN"I",1,B$:OPEN"O",2,BP$:REM ABRE O ARQUIVO SEQUENCIAL B
$ ( PROGRAMA A SER PROTEGIDO ), PARA LEITURA, PELO BUFFER 1.ABRE
 O ARQUIVO SEQUENCIAL QUE SERA O PROGRAMA PROTEGIDO COM O NOME B
P$ PELO BUFFER 2, PARA GRAVACAO.
20040 IF LOC(1) THEN CLOSE:KILL B$:END:REM SE TIVER ALCANCADO O
FIM DO PROGRAMA QUE ESTA SENDO PROTEGIDO ENTAO FECHE OS ARQUIVOS
 E MATE O PROGRAMA ASCII QUE FOI CONVERTIDO (AGORA DESNECESSARIO
).
20041 REM A PARTIR DESSE PONTO O USUARIO DEVERA FAZER LOAD AO PR
OGRAMA PROTEGIDO (COM SUFIXO "/PRT") E LOGO DEPOIS FAZER SAVE, C
OM O MESMO NOME COM SUFIXO "/PRT", DE FORMA A RECONVERTER O PROT
EGIDO (ASCII) EM UM NORMAL (TOKENS).
20050 LINE INPUT#1,P$:REM LE UMA LINHA DE PROGRAMA DO PROGRAMA E
M FORMATO ASCII A SER PROTEGIDO E COLOCA EM P$
20060 P$=P$+A$:REM FAZ A COMPOSICAO DA STRING, UNINDO A LINHA (E
M P$) COM A$ (CARACTERES DE PROTECAO).
20070 PRINT#2,P$:REM ENVIA PELO BUFFER 2 PARA O NOVO ARQUIVO (PR
OGRAMA PROTEGIDO) O NOVO CONTEUDO DA LINHA (EM P$) AGORA PROTEGI
DA PELO PROGRAMA PROTECTOR.
20080 GOTO 20040:REM REINICIA O LACO.
```

Protector

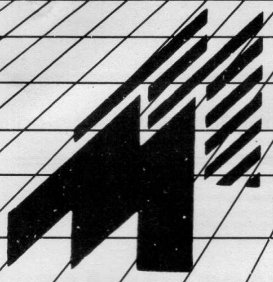
ISBL DADOS®
sistemas de banco de dados

CURSOS DISPONÍVEIS

- Introdução à Microcomputação
- DOS – PC "Sistema Operacional"
- * UNIX "Sistema Operacional"
- LINGUAGEM C "Ling. Programação"
- dBASE II "Programação Básica"
- dBASE II "Program. Avançada"
- dBASE III "Program. Básica"
- LOTUS 1-2-3 "Plan. Eletrônica"
- * Framework "Sistema Integrado"
- Symphony "Sistema Integrado"
- Wordstar "Processador de Texto"

* Em desenvolvimento

MATERIAIS DIDÁTICOS: Publicações Técnicas desenvolvidas em português.
RECURSOS DIDÁTICOS: Conceitos e exemplos práticos, através de Micros e Telão de 72"
CURSOS FECHADOS E ABERTOS
NOVO ENDEREÇO
Rua Groelândia, 1.750 - Jardim Europa
CEP - 01434 - São Paulo - SP
Telefones: 881-4599 - 881-4032



MICROMAQ

Sempre Novidades

Atenção Usuários de Color Computer

(CP-400, Color 64, TRS-80 Color, etc.)

Você já conhece nosso Catálogo 86?

São mais de 300 produtos entre livros, hardware, software, suprimentos, etc, para seu micro.

Escreva-nos solicitando seu exemplar, INTEIRAMENTE GRÁTIS, e mantenha-se atualizado com o mundo do color.

TAPEDUPE

Duplicador de fitas para COLOR. Duplica qualquer fita, mesmo com capa de proteção.

PU 021 F ..... 3,5 ORTN

Baixou!

COLOR MAX

— novo preço —

(O melhor programa de Gráficos para COLOR) Composto de Placa de Hardware e programa em fita ou disco:

E 001 F - Fita ..... 8,8 ORTN
E 001 D - Disco ..... 11,7 ORTN

MSX Já temos programas para o MSX. Escreva ou telefone solicitando informações.

Faça seu pedido pelo correio, para o endereço abaixo, anexando Cheque ou Vale Postal e receba, SEM MAIS DESPESAS, em aproximadamente 15 dias.

MICROMAQ - Rua Sete de Setembro, 92 - Loja 106 - Tel.: 222.6088 - Rio de Janeiro, RJ - CEP 20050

Mizael

Solitário

Augusto Anderson Brown Santos

Solitário foi adaptado para a linha TRS-Color com um mínimo de 16Kb de RAM, a partir de um programa de mesmo nome publicado em **MS nº 42** (a primeira versão, compatível com a linha Sinclair, é de autoria de Roberto Ribeiro Peixinho).

O programa apresenta-se incrementado com cores e som, além de possuir uma abertura com meu logotipo, que pode ser omitida.

O jogo em si é o mesmo, tendo sido feitas, porém, as seguintes modificações:

- Após formada a tela, deve-se digitar as coordenadas e, quando aparecer a palavra < INTRODUZIR >, deve-se digitar <S> ou <N>.
- Quando não houver mais chances de eliminar peças, deve-se digitar <X> para obter a classificação.
- O programa trabalha em alta velocidade – POKE 65495,0 – requerendo que o computador esteja em velocidade normal na hora da gravação.

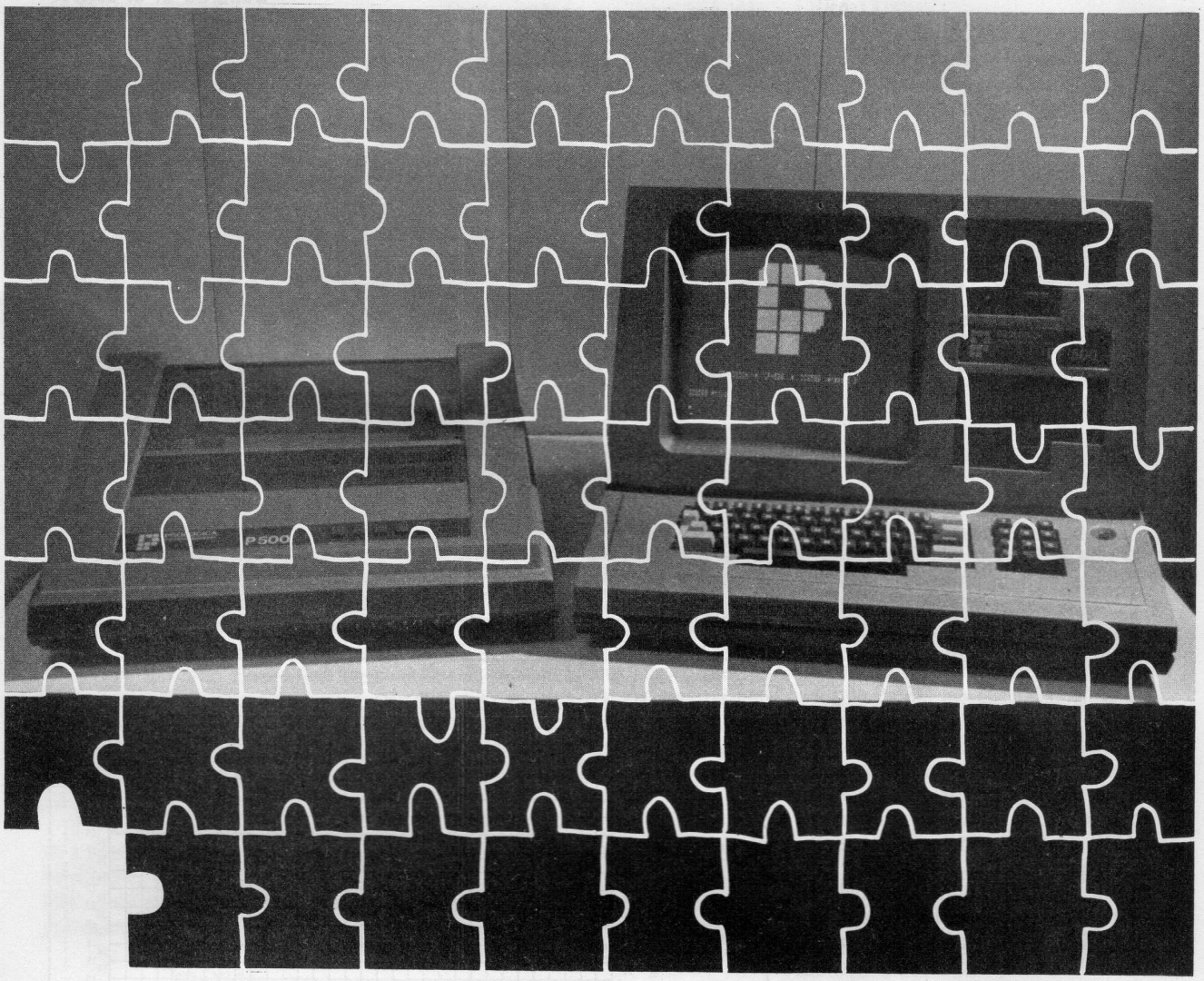
Augusto A. B. Santos está cursando a 3ª série do segundo grau e estuda BASIC. Possui um CP 400 e um CP 200, há um ano e meio.

```
1 PMODE4,1:SCREEN1,1:PCLS
2 DRAW"BM0,70;E11;R233;F11;BM0,1
77;F11;R233;E11"
3 PAINT(0,191),5,5
4 DRAW"BM31,11;D24;F16;R4;U12;B;
U4;U4;R8;D4;H4;F4;L8;B;D4;R8;D12
;H7;F7;R4;U24;H16;L16;F16;R16;L1
6;D24"
5 DRAW"BM55,11;D7;U7;F16;D24;H7;
F7;R16;U24;H16;L4;D7;U7;F16;R4;L
4;D20;H7;F7;L8;U20;L4;R4;H16;L4"

6 DRAW"BM79,11;D7;U7;F16;D24;H7;
F7;R16;U12;H7;F7;L8;H3;F3;D4;H3;
F3;R4;D4;L8;U16;R12;U4;L16;R16;H
16;L16"
7 DRAW"BM103,11;D7;U7;F16;D24;H7
;F7;R16;U24;H16;L4;D7;U7;F16;R4;
L4;D20;H7;F7;L8;U20;L4;R4;H16;L4
"
8 DRAW"BM127,11;D7;U7;F16;D14;H7
;F7;R12;D6;H6;F6;L12;H7;F7;D4;H7
;F7;R16;U14;H6;F6;L12;U6;R12;U4;
L16;R16;H16;L16"
9 DRAW"BM151,11;D6;U2;F16;U4;D4;
R6;D20;H16;F16;R4;U20;R6;U4;H16;
L16;F16;R16"
10 DRAW"BM175,11;D7;U7;F16;D24;H
13;F13;R4;B;U4;U16;R8;D16;H7;F7;
L8;B;D4;R12;U24;L16;R16;H16;L16"

11 POKE178,1:PAINT(0,0),,5:POKE1
78,3
12 DRAW"BM60,80;L16;D12;R16;D12;
L16;BM84,80;L16;D24;R16;U24;BM10
8,80;L16;D24;U12;R12;BM136,80;L2
0;R10;D24;BM144,80;D24;E9;F9;U24
;BM172,104;U24;R16;D24;U12;L16;B
M144,138;U24;R16;D12;L16;F12;BM1
88,114;L16;D12;R12;L12;D12;R16"
13 DRAW"BM67,124;L7;D7;R7;BM70,1
31;U7;R7;D3;L7;BM84,124;D3;R7;U3
;D7;BM94,124;D7;R7;U7;L7;BM104,1
24;D7;R7;U7;L7"
14 DRAW"BM80,180;L26;E13;F13":LI
NE(80,180)-(87,173),PSET:LINE(87
,173)-(67,167),PSET
15 DRAW"BM182,180;L26;E13;F13":L
INE(182,180)-(189,173),PSET:LINE
(189,173)-(169,167),PSET
16 DRAW"BM132,160;L26;E13;F13":L
INE(132,160)-(139,153),PSET:LINE
(139,153)-(119,147),PSET
17 DRAW"BM95,180;U7;R4;D3;L4;F4;
BM107,173;L4;D4;R4;L4;D3;R4;BM11
5,173;L5;D3;R5;D4;L5;BM126,173;L
8;R4;D7;BM129,180;U7;R4;D4;L4;R4
;D3;BM136,177;R4;BM143,180;R4;L2
;U7;L2"
18 FORA=1TO6:PLAY"O4;T100;BCAE":
NEXT
19 FORA=1TO4500:NEXT
100 GOSUB840
110 CLS0:POKE65495,0
120 FORN=1TO16:PRINT@N,CHR$(163)
;:PRINT@480+N,CHR$(172);:NEXT
130 FORM=1TO14:PRINT@M*32+1,CHR$
(175);:PRINT@M*32+16,CHR$(175);:
NEXT
140 FORN=1TO3:FORM=2TO4:PRINT@N*
32+M,CHR$(166);:PRINT@N*32+M+11,
CHR$(169);:PRINT@(N+11)*32+M,CHR
$(169);:PRINT@(N+11)*32+M+11,CHR
$(166);:NEXTM,N
150 FORN=1TO4:PRINT@N*32+5,CHR$(
167);:PRINT@N*32+12,CHR$(171);:P
RINT@(N+10)*32+5,CHR$(173);:PRIN
T@(N+10)*32+12,CHR$(174);
160 PRINT@129+N,CHR$(167);:PRINT
@139+N,CHR$(171);:PRINT@353+N,CH
R$(173);:PRINT@363+N,CHR$(174);:
NEXT
170 FORM=2TO14STEP2:PRINT@M,CHR$
(97+(M-2)/2);:PRINT@480+M,CHR$(9
7+(M-2)/2);:PRINT@(M-1)*32+0,CHR
$(49+(M-2)/2);:PRINT@(M-1)*32+17
,CHR$(49+(M-2)/2);:NEXT
180 DIM S(2),P$(3),T(7,7),J$(4)
190 P$(1)=CHR$(145)+CHR$(146)
200 P$(2)=CHR$(148)+CHR$(152)
210 P$(3)=STRING$(2,128)
220 FORN=5TO9STEP2:FORM=2TO14STE
P2:PRINT@N*32+M,P$(1);:PRINT@(N+
1)*32+M,P$(2);:NEXTM,N
230 FORN=1TO3STEP2:FORM=6TO10STE
P2:PRINT@N*32+M,P$(1);:PRINT@(N+
1)*32+M,P$(2);:PRINT@(N+10)*32+M
,P$(1);:PRINT@(N+11)*32+M,P$(2);
:NEXTM,N
240 PRINT@232,P$(3);:PRINT@264,P
$(3);
250 PRINT@22,"pecas";:PRINT@53,"
em";CHR$(128);"jogo";
260 P=32
270 PRINT@86,CHR$(225);STRING$(4
,227);CHR$(226);:PRINT@118,CHR$(
229);STRING$(4,128);CHR$(234);:P
RINT@119,P;:PRINT@150,CHR$(228);
STRING$(4,236);CHR$(232);
280 PRINT@182,"pecas";:PRINT@212
,"retiradas";
290 R=0
300 PRINT@246,CHR$(225);STRING$(
4,227);CHR$(226);:PRINT@278,CHR$
(229);STRING$(4,128);CHR$(234);:
PRINT@279,R;:PRINT@310,CHR$(228)
;STRING$(4,236);CHR$(232);
310 PRINT@339,"sua";CHR$(128);"j
ogada";:PRINT@372,"de";STRING$(3
,128);"para";
320 PRINT@403,CHR$(209);STRING$(
2,211);CHR$(210);STRING$(2,128);
CHR$(209);STRING$(2,211);CHR$(21
0);:PRINT@435,CHR$(213);STRING$(
2,128);CHR$(218);STRING$(2,128);
CHR$(213);STRING$(2,128);CHR$(21
8);
330 PRINT@467,CHR$(212);STRING$(
2,220);CHR$(216);STRING$(2,128);
CHR$(212);STRING$(2,220);CHR$(21
6);
340 S(1)=0:S(2)=1
350 FORX=1TO7:FORY=1TO7:T(X,Y)=S
(2):NEXTY,X
360 T(4,4)=S(1)
370 FORW=1TO31
380 FORJ=1TO2
390 A$=INKEY$:IFA$=""THEN390
400 IFA$="X"THEN890
410 IFASC(A$)<650R ASC(A$)>71THE
N390
420 SOUND50,3:PRINT@436+(J-1)*6,
CHR$(ASC(A$));:J$(2*J-1)=A$
430 A$=INKEY$:IFA$=""THEN430
440 IFA$="X"THEN890
450 IFASC(A$)<490R ASC(A$)>55THE
N430
460 SOUND100,3:PRINT@437+(J-1)*6
,CHR$(ASC(A$));:J$(2^J)=A$
470 NEXTJ
480 PRINT@499,"introduzir";
490 A$=INKEY$:IFA$=""THEN490
500 IFA$="S"THEN530
510 IFA$()"N"THEN490
520 PRINT@436,STRING$(2,128);:PR
INT@442,STRING$(2,128);:PRINT@49
9,STRING$(11,128);:GOTO380
530 PRINT@499,STRING$(11,128);
540 C=ASC(J$(1))-64
550 L=ASC(J$(2))-48
560 C=ASC(J$(3))-64
570 I=ASC(J$(4))-48
580 IF L<>I AND C<>0 THEN810
590 IFL(=2ANDC(=20RI(=2ANDO(=2TH
EN810
600 IFL(=2ANDC)=60RI(=2ANDO)=6TH
EN810
610 IFL)=6ANDC)=60RI)=6ANDO)=6TH
EN810
620 IFL)=6ANDC(=20RI)=6ANDO(=2TH
EN810
630 IF L=I AND(C+0)/2<>INT((C+0)
/2)THEN810
640 IFC=0AND(L+I)/2<>INT((L+I)/2
)THEN810
650 IFL=I THEN H=L:A=(C+0)/2
660 IFC=0 THEN A=C:H=(L+I)/2
670 IFT(L,C)=0THEN810
680 IFT(H,A)=0THEN810
690 IFT(I,0)=1THEN810
700 PRINT@(2*L-1)*32+2*C,P$(3);:
PRINT@(2*L)*32+2*C,P$(3);
710 PRINT@(2*I-1)*32+2*0,P$(1);:
PRINT@(2*I)*32+2*0,P$(2);
720 PRINT@(2*H-1)*32+2*A,P$(3);:
PRINT@(2*H)*32+2*A,P$(3);
730 T(L,C)=S(1)
740 T(H,A)=S(1)
750 T(I,0)=S(2)
760 R=R+1:P=P-1
770 PRINT@3*32+23,STRING$(4,128)
;:PRINT@3*32+23,P;:PRINT@8*32+23
,R;
780 PRINT@436,STRING$(2,128);:PR
INT@442,STRING$(2,128);
790 NEXTW
800 CLS:PRINT@130,"SINTO MUITO M
AS SUAS CHANCES",,,"TERM
INARAM.":SCREEN0,1:FORN=1TO4000
:NEXT:GOTO890
810 PRINT@500,"invalida";:PLAY"V
30;O2;L2.;T30;ACDGD":FORN=1TO100
0:NEXT
820 PRINT@500,STRING$(8,128);:PR
INT@436,STRING$(2,128);:PRINT@44
2,STRING$(2,128);
830 GOTO790
840 CLS:PRINTSTRING$(11,163);"SO
LITARIO";STRING$(12,163);STRING$
(32,172);
850 PRINT@76,"regras","1-)VOCE D
EVE TENTAR ELIMINAR PECAS,
SOBRANDO APENAS *UMA*.","2-)A PE
CA JOGADA DEVE PULAR A PECA
VIZINHA E CAIR NUMA CASA
VAZIA.",,"3-)NAO VALE JOGAR NAS
DIAGONAIS.";"4-)SE NAO HOUVER M
AIS CHANCE DE ";
860 PRINT" ELIMINAR PECAS TECL
E (X) PARA SABER SUA CLASSIFIC
ACAO.","5-)VOCE TEM 31 JOGADAS,I
NCLU- INDO AS INVALIDADAS."
,"  B O A  S O R T E . . ."
870 PRINT@482,"QUALQUER TECLA PA
RA INICIAR";:SCREEN0,1
880 EXEC44539:RETURN
890 CLS:PRINT"POIS E',",,,
900 IFP)=6THENPRINT"LAMENTAVEL,V
OCE FOI UM fracasso;AINDA SOBRAR
AM";P;"PECAS."
910 IFP=5THENPRINT"VOCE ESTA fra
co MAS TEM CHANCE DE MELHORAR.R
ESTARAM";P;"PECAS."
920 IFP=4THENPRINT"VODE ESTA reg
ular MAS PODE MELHORAR SUA
CLASSIFICACAO.","RESTARAM";P;"PE
CAS."
930 IFP=3THENPRINT"VOCE E'bom NI
STO.DASAFIO-0 A MELHORAR...",
,"RESTARAM";P;"PECAS."
940 IFP=2THENPRINT"VOCE E' otimo
.POR POUCO NAO CONSEGUIU.RES
TARAM APENAS";P,"PECAS."
950 IFP=1THENPRINT"FANTASTICO! V
OCE E' excelente, SEM DUVIDAS O
CAMPEAO.RESTOU APENAS";P;"PE
CA.TENTE REPETIR O FATO."
960 PRINT@354,"QUER JOGAR NOVAME
NTE ?(S/N)":SCREEN0,1
970 A$=INKEY$:IFA$=""THEN970
980 IFA$="S"THENRUN110
990 IFA$()"N"THEN970ELSEEND
1000 ##############################
     #      solitario      #
     #ROBERTO RIBEIRO PEIXINHO #
     #ADAPTADO PARA O COLOR POR#
     #AUGUSTO ANDERSON BROWN S.#
     ##############################
```

Solitário

SUPRIMENTOS PARA CPD FILCRES:

SOLUÇÕES NA MEDIDA EXATA DE SUAS NECESSIDADES.



A Filcres tem sempre em estoque e para pronta entrega tudo o que você precisa para o seu CPD: fitas impressoras, formulários, diskettes, e mais uma infinidade de outros acessórios.
E conta ainda com uma equipe de profissionais altamente qualificados, prontos a lhe prestar o melhor atendimento,
Tudo isso com a qualidade que v. exige e um preço sempre compatível com seu orçamento.
É só telefonar e esquecer seu problema.
A Filcres leva o suprimento que falta até você.

20 Anos de Sucesso

FILCRES

AJUDANDO A DESENVOLVER TECNOLOGIA

Rua Aurora, 165 – SP – Tel.: PBX (011) 223-7388 – Telex (011) 31298 Direto – SP – 223-1446/ 222-3458/220-9113/220-7954 Direto outros Estados: 222-5430/221-0326/223-7649/222-0284

iniciante

Neste artigo você entenderá como manipular a tela dos micros das linhas Apple e Sinclair.

Eduardo O. C. Chaves

Manipulando a tela

No artigo *A tela do seu micro*, publicado em MS nº 49 (Seção Iniciante), descrevemos apenas algumas das principais características da tela do computador. Agora vamos começar a falar sobre como se manipula, usando a linguagem BASIC, a tela dos principais tipos de microcomputadores: os das famílias Sinclair e Apple. Em outra oportunidade, falaremos sobre a dos TRS-80, Mod III.

A tela, tanto no modo de baixa como no de alta resolução, deve ser entendida, como vimos no artigo anterior, em termos de uma série de linhas e de colunas, que, ao se interceptarem, formam uma coordenada 1,c (poderia ser x,y). Cada coordenada corresponde a um ponto de interseção de uma linha (horizontal) com uma coluna (vertical) da tela.

Embora os microcomputadores de todas essas famílias utilizem a linguagem BASIC, o dialeto BASIC implementado em cada uma delas sempre apresenta diferença em relação ao dialeto implementado nas outras. Por causa disso, vamos ter que lidar não só com os diferentes tipos de tela, mas também com diferentes tipos de instruções para manipulá-las.

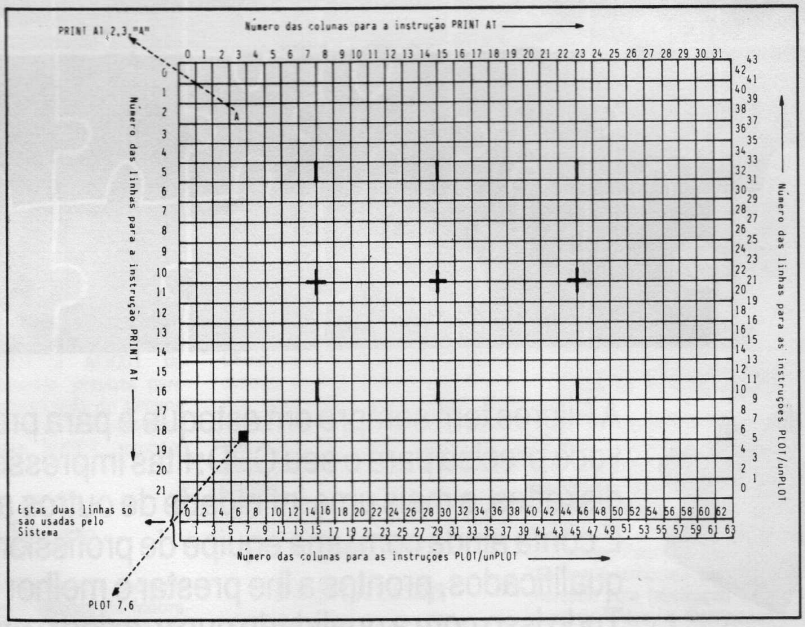


Figura 1 – Tela dos microcomputadores da linha Sinclair.

FAMÍLIA SINCLAIR

Vamos começar com os micros da família Sinclair que, no Brasil, são representados principalmente pelo TK-82, 83 e 85 e pelo CP-200. Com esses micros, você pode imprimir texto a partir de qualquer posição predeterminada da tela, usando a instrução **PRINT AT linha, coluna; cadeia de caracteres ou número.**

Não se esqueça de que os micros dessa família têm 22 linhas e 32 colunas utilizáveis pelo usuário. Os números das linhas vão de 0 a 21; e os das colunas, de 0 a 31 (veja a figura 1). Se você usar a instrução **PRINT AT 0,0;"*"**, seu micro vai imprimir um asterisco na linha 0 e na coluna 0 da tela. Mas qual é a linha 0? É a primeira linha horizontal em cima ou embaixo? E qual é a coluna 0? É à esquerda ou à direita?

Para a instrução PRINT AT, a linha 0 é a linha que fica na parte mais de cima da tela, e a linha 21 a que fica na parte mais de baixo. (Como vimos no outro artigo, a tela dos micros da família Sinclair ainda possui mais duas linhas, as de número 22 e 23, mas você não pode imprimir nelas: elas são usadas para mensagens do sistema). No tocante às colunas, não há muita margem para dúvidas: a coluna 0 é a que fica

mais à esquerda; e a coluna 31, a que fica mais à direita da tela. A instrução que usamos acima, portanto, vai imprimir um asterisco no canto superior esquerdo da tela.

Dissemos, ao final do outro artigo, que os micros da linha Sinclair não eram adequados para gráficos de alta resolução. Na verdade não são. Mas o BASIC dessa linha de micros possui instruções que permitem a definição de gráficos de média-baixa resolução. Estas instruções são PLOT e UNPLOT.

Quando as instruções PLOT e UNPLOT são utilizadas, a tela parece que dobra de tamanho, pois estamos agora lidando com 44 linhas e 64 colunas – na verdade, ela parece quadruplicar de tamanho, pois dobra o número de linhas e o de colunas. Só que tanto as linhas como as colunas têm seu tamanho diminuído pela metade. Dessa forma, o que aparece na tela, quando usamos a instrução PLOT, é um quadradinho (na verdade um pequeno retângulo) que tem basicamente um quarto do tamanho de um caráter normal. Então, dentro do espaço que é ocupado por um caráter normal, podem agora ser plotados quatro pontos, e, assim, as 704 posições, que podem ser manipuladas na tela de baixa resolução (22 x 32), agora aumentam para 2.816 (44 x 64), isto é, quadruplicam em número, porque agora possuem um quarto do tamanho de um caráter normal. Foi por isso que chamamos essa resolução de média-baixa resolução: 2.816 não é nada que se possa comparar com os 53.760 pontos de alta resolução dos micros da família Apple. Mas, por outro lado, é quatro vezes mais que as 704 posições da baixa resolução.

Você já deve ter descoberto que a instrução PLOT coloca um quadradinho na tela, em uma dada posição, e a instrução UNPLOT remove o quadradinho que porventura houver na posição especificada. O modelo para instrução PLOT – e o da instrução é idêntico – é o seguinte: **PLOT coluna, linha**. Há duas diferenças básicas e importantes entre as instruções PRINT AT, de um lado, e PLOT e UNPLOT, de outro. Uma delas é que, no caso das instruções PLOT e UNPLOT, o primeiro parâmetro – primeiro número que aparece depois da instrução – se refere à coluna, e o segundo à linha, enquanto que no caso da instrução PRINT AT, como vimos, é o oposto.

A segunda diferença é que, para as instruções PLOT e UNPLOT, as linhas vão de 0 a 43 e as colunas de 0 a 63, sendo a linha 0 a primeira linha da parte de baixo, e a coluna 0 a primeira coluna à esquerda da tela. Na instrução PRINT AT a linha 0 é a primeira linha na parte de cima da tela.

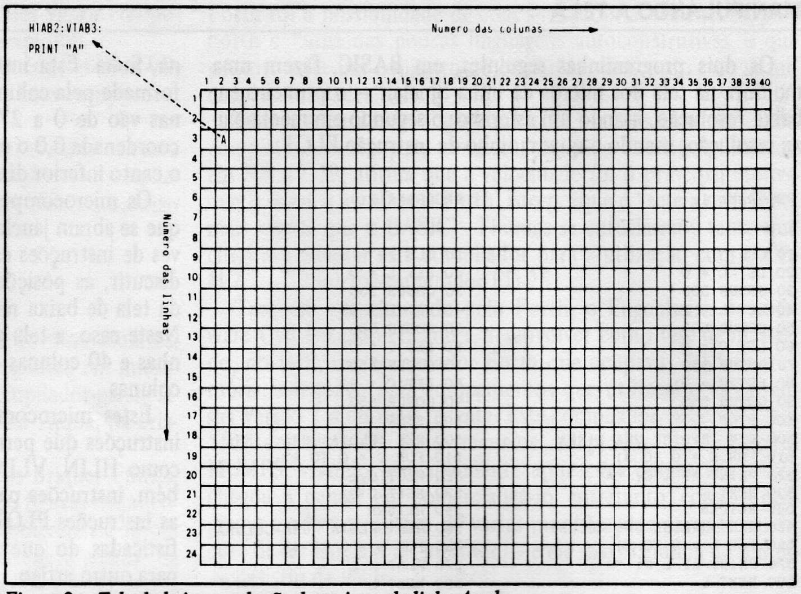


Figura 2 – Tela de baixa resolução dos micros da linha Apple.

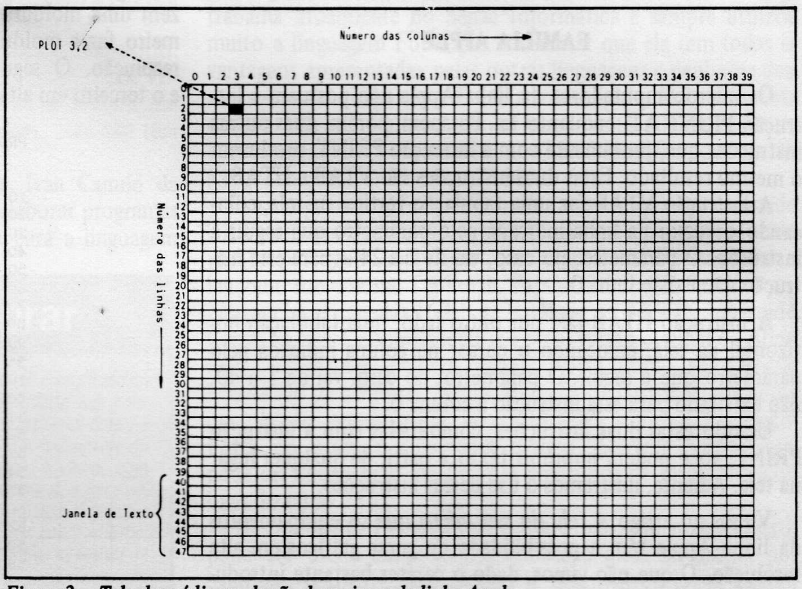


Figura 3 – Tela de média resolução dos micros da linha Apple.

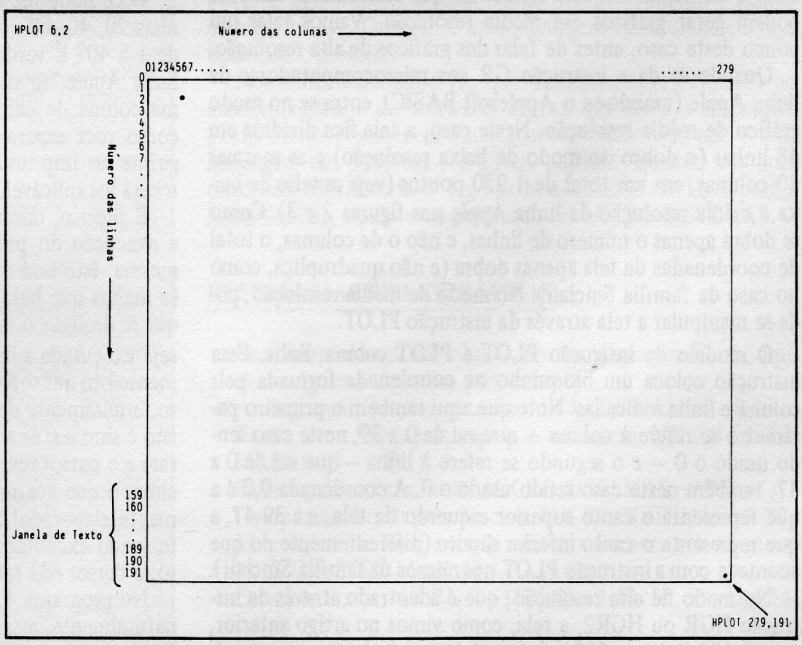


Figura 4 – Tela de alta resolução dos micros da linha Apple (as coordenadas aqui representam porções pequenas da tela que não dá para quadricular).

Os dois programinhas seguintes, em BASIC, fazem uma moldura na tela dos micros da linha Sinclair – o primeiro em baixa resolução, usando um asterisco; o segundo em média-baixa resolução, usando o quadradinho da instrução PLOT:

PROGRAMA 1:

```
10 CLS
20 LET L = 0
30 GOSUB 500
40 LET L = 21
50 GOSUB 500
60 LET C = 0
70 GOSUB 550
80 LET C = 31
90 GOSUB 550
100 STOP
500 REM IMPRIME UMA LINHA
510 FOR C = 0 TO 31
520 PRINT AT L,C; "*"
530 NEXT C
540 RETURN
550 REM IMPRIME UMA COLUNA
560 FOR L = 0 TO 21
570 PRINT AT L,C; "*"
580 NEXT L
590 RETURN
```

PROGRAMA 2:

```
10 CLS
20 LET L = 0
30 GOSUB 500
40 LET L = 43
50 GOSUB 500
60 LET C = 0
70 GOSUB 550
80 LET C = 63
90 GOSUB 550
100 STOP
500 REM 'PLOTA' UMA LINHA
510 FOR C = 0 TO 63
520 PLOT C,L
530 NEXT C
540 RETURN
550 REM 'PLOTA' UMA COLUNA
560 FOR L = 0 TO 43
570 PLOT C,L
580 NEXT L
590 RETURN
```

FAMÍLIA APPLE

Os microcomputadores da linha Apple não possuem a instrução PRINT AT, enquanto tal. Possuem, porém, duas outras instruções que, juntamente com a instrução PRINT, produzem o mesmo resultado. Estas duas instruções são VTAB e HTAB.

A instrução VTAB faz uma tabulação vertical na tela, colocando o cursor na linha indicada pelo parâmetro que seguir a instrução. O parâmetro, no caso, vai de 1 a 24 – para esta instrução não existe linha 0.

A instrução HTAB faz, por outro lado, uma tabulação horizontal na tela, colocando o cursor na coluna indicada pelo parâmetro que a seguir. O parâmetro, no caso, vai de 1 a 40, não existindo para esta instrução a coluna 0.

Usando estas duas instruções, combinadas com a instrução PRINT, você poderá imprimir texto a partir de qualquer lugar na tela. Adiante, ilustramos o uso dessas instruções.

Vimos no artigo *A tela do seu micro* que os equipamentos da linha Apple têm a possibilidade de gerar gráficos em alta resolução. O que não vimos, dado o caráter bastante introdutório e elementar daquele artigo, foi que estes micros também podem gerar gráficos em média resolução. Vamos falar um pouco deste caso, antes de falar dos gráficos de alta resolução.

Quando se dá a instrução GR aos microcomputadores da linha Apple (usando-se o Applesoft BASIC), entra-se no modo gráfico de média resolução. Neste caso, a tela fica dividida em 48 linhas (o dobro do modo de baixa resolução) e as mesmas 40 colunas, em um total de 1.920 pontos (veja as telas de baixa e média resolução da linha Apple nas figuras 2 e 3). Como se dobra apenas o número de linhas, e não o de colunas, o total de coordenadas da tela apenas dobra (e não quadruplica, como no caso da família Sinclair). No modo de média resolução, pode-se manipular a tela através da instrução PLOT.

O modelo da instrução PLOT é **PLOT coluna, linha**. Essa instrução coloca um bloquinho na coordenada formada pela coluna e linha indicadas. Note que aqui também o primeiro parâmetro se refere à coluna – que vai de 0 a 39, neste caso sendo usado o 0 – e o segundo se refere à linha – que vai de 0 a 47, também neste caso sendo usado o 0. A coordenada 0,0 é a que representa o canto superior esquerdo da tela, e a 39,47, a que representa o canto inferior direito (diferentemente do que acontece com a instrução PLOT nos micros da família Sinclair).

No modo de alta resolução, que é adentrado através da instrução HGR ou HGR2, a tela, como vimos no artigo anterior, possui 192 linhas e 280 colunas. A instrução básica para manipular a tela, neste caso, é HPLOT, cujo modelo é **HPLOT coluna, linha**. Esta instrução coloca um pontinho na coordenada formada pela coluna e pela linha indicadas. Neste caso, as colunas vão de 0 a 279 e as linhas de 0 a 191, representando a coordenada 0,0 o canto superior esquerdo da tela, e a 279,191, o canto inferior direito (veja a figura 4).

Os microcomputadores da linha Apple também permitem que se abram janelas de texto nas telas de alta resolução. Através de instruções específicas, que aqui não vem tanto ao caso discutir, as posições correspondentes às últimas quatro linhas da tela de baixa resolução podem ficar disponíveis para texto. Neste caso, a tela gráfica de média resolução ficaria com 40 linhas e 40 colunas, e a de alta resolução com 160 linhas e 280 colunas.

Estes microcomputadores têm ainda uma série de outras instruções que permitem traçar linhas e outras figuras na tela, como HLIN, VLIN, DRAW, XDRAW, SCALE etc. Há, também, instruções para alterar as cores dos gráficos. Além disso, as instruções PLOT e HPLOT também têm utilizações mais sofisticadas do que as que indicamos aqui. Mas isto é assunto para outro artigo.

Logo abaixo vai o esqueleto de três programinhas que fazem uma moldura na tela dos micros da linha Apple. O primeiro faz a moldura usando um asterisco, em modo de baixa resolução. O segundo faz a moldura em média resolução, e o terceiro em alta.

PROGRAMA 3

```
10 HOME
20 X = 1 : Y = 39
30 GOSUB 500
40 X = 23
50 GOSUB 500
60 Y = 1 : X = 23
70 GOSUB 550
80 Y = 39
90 GOSUB 550
95 GOTO 95
100 END
500 REM IMPRIME UMA LINHA
510 FOR I = 1 TO Y
520 VTAB X : HTAB I : PRINT "*"
530 NEXT I
540 RETURN
550 REM IMPRIME UMA COLUNA
560 FOR I = 1 TO X
570 HTAB Y : VTAB I : PRINT "*"
580 NEXT I
590 RETURN
```

Você pode ter achado estranho utilizarmos 23 e 39 nas linhas 20, 40, 60 e 80. As linhas não vão de 1 a 24 e as colunas de 1 a 40? É verdade. Só que, com os microcomputadores da linha Apple, se você chegar a imprimir alguma coisa na última coluna de cada linha, a tela do computador não vai ficar como você espera. Isto porque o cursor passa para a linha seguinte ao imprimir na última coluna – e o resultado lhe parecerá inexplicável. Experimente.

É preciso, também, explicar a linha 95. Ela determina que a execução do programa volte para a linha 95, isto é, para si mesma. Isto fará com que o programa fique interminavelmente (a menos que haja interrupção pelo teclado, ou, naturalmente, que se desligue o micro) executando a linha 95, sem que jamais seja executada a linha 100 que encerra o programa – e é exatamente isto que se pretende. A linha 100, neste programa, é, portanto, inteiramente desnecessária: ela não é executada. A razão para isto é simples: se a linha 100 for executada, o programa se encerrará e o cursor reaparecerá, fazendo com que a tela se mova para cima, o que acarretará a destruição da moldura tão cuidadosamente elaborada! Se a linha 95, porém, ficar interminavelmente sendo executada, o programa nunca se encerrará, e, portanto, o cursor não reaparecerá para destruir a moldura.

No programa 4, o computador está programado para exibir, naturalmente, a janela de oito linhas por 40 colunas embaixo. Por isso usamos 39 nas linhas 40 e 560. Se você quiser alterar o valor nessas linhas para 47, precisará acrescentar duas linhas

ao programa: uma linha 11, em que se inclua a instrução **CALL -1998**, e uma linha 12, onde se inclua a instrução **POKE -16302,0**. E deverá ainda alterar o conteúdo da linha 95 para GOTO 95, para que o final da execução do programa não destrua a sua moldura. Experimente.

PROGRAMA 4

```
5 HOME
10 GR
15 COLOR = 15
20 L = 0
30 GOSUB 500
40 L  = 39 (47)
50 GOSUB 500
60 C = 0
70 GOSUB 550
80 C = 39
90 GOSUB 550
95 HTAB 1 : VTAB 21
100 END
500 REM 'PLOTA' UMA LINHA
510 FOR C = 0 TO 39
520 PLOT C,L
530 NEXT C
540 RETURN
550 REM 'PLOTA' UMA COLUNA
560 FOR L = 0 TO 39 (47)
570 PLOT C,L
580 NEXT L
590 RETURN
```

PROGRAMA 5

```
5 HOME
10 HGR2 (HGR)
15 HCOLOR = 7
20 L = 0
30 GOSUB 500
40 L = 191 (159)
50 GOSUB 500
60 C = 0
70 GOSUB 550
80 C = 279
90 GOSUB 550
95 GOTO 95
100 END
500 REM 'PLOTA' UMA LINHA
510 FOR C = 0 TO 279
520 HPLOT C,L
530 NEXT C
540 RETURN
550 REM 'PLOTA' UMA COLUNA
560 FOR L = 0 TO 191 (159)
570 HPLOT C,L
580 NEXT L
590 RETURN
```

No programa 5, se você usar HGR2 na linha 10, não precisará acrescentar nenhuma instrução – o texto aparecerá sem janela. Se usar HGR, porém, o texto aparecerá com uma janela. Neste caso, use os valores entre parênteses nas linhas 40 e 560. Então, o conteúdo da linha 95 poderá ser **HTAB 1 : VTAB 21**, pois esta instrução colocará o cursor dentro da janela de texto, antes do encerramento do programa.

Os programas que fizemos não são os mais eficientes, ou os mais bem estruturados. Nossa intenção foi basicamente fazer programas fáceis de se entender. Aqui vai um programa bem menor que faz a mesma coisa que o programa 4, com a janela:

PROGRAMA 6:

```
5 HOME
10 GR
20 COLOR = 15
30 FOR A = 0 TO 39
40 PLOT A,0 : PLOT A,39
50 PLOT 0,A : PLOT 39,A
60 NEXT A
```

Tente entender por que os dois programas realizam basicamente a mesma coisa. Experimente também fazer outras coisas, além de molduras. Ninguém vai querer fazer molduras na tela a vida inteira, não é mesmo? Mas posso garantir que, se você souber fazer molduras, conseguirá, com um pouco de trabalho, fazer qualquer outro desenho na tela. É só encontrar as coordenadas certas.

Bom, vamos parando por aqui. Futuramente, discutiremos outras linhas de equipamento.

Eduardo O. C. Chaves é Coordenador do Centro de Informática Aplicada da UNICAMP e Consultor Editorial da PEOPLE Computação, de Campinas, SP.

Mesmo no escuro, o programa vai ficar na memória.

Utilizando UNIVOLT nos micro-computadores de sua empresa, a falta de energia jamais interromperá o trabalho eliminando os riscos com a perda da memória.
UNIVOLT é o gerador eletrônico mais prático, portátil e que dispensa instalações especiais.
UNIVOLT O PAR PERFEITO PARA OS MICROS.

Unitron
engenharia, indústria e comércio ltda.
Rua Antonieta Leitão, 110 - Freguesia do Ó - SÃO PAULO - SP
PABX - (011) 858-4744 - TX. (011) 32003 UEIC-BR

REPRESENTANTES: • São Paulo (011) 858-4744 • Campinas (0192) 2-1575 • Rio de Janeiro (021) 201-7643 • Porto Alegre (0512) 22-5288 • Curitiba (041) 262-8423 • Blumenau (0473) 22-4422 • Belo Horizonte (031) 225-6425 • Brasília (061) 248-5359 • Goiânia (062) 224-7271 • Salvador (071) 245-7812 • Aracajú (079) 224-7776 • Maceió (082) 221-2617 • Recife (081) 325-3189 • Belém (091) 222-5122.

Nesta lição, discutiremos os recursos que FORTH oferece para se montar estruturas de dados. Depois, é só usar a imaginação!

Curso de FORTH (V)

Antonio Costa

As estruturas de dados são consideradas os pilares da computação e não é possível programar bem sem usá-las. Se você ainda não sabe nada a respeito de estruturas de dados, não posso deixar de recomendar-lhe que leia um dos muitos livros escritos sobre o assunto.

Os gregos escreviam em folhas de papiro, as quais eram enroladas e jogadas num canto qualquer. A coisa funcionou bem enquanto o número de obras disponíveis era pequeno. Quando a quantidade de folhas que aquele culto povo possuia aumentou muito, alguém teve a idéia de encaderná-las e, assim, surgiu uma estrutura para armazenar dados chamada livro. A produção intelectual dos gregos era vastíssima e, em pouco tempo, os próprios livros se tornaram por demais numerosos. Isto levou um rei de Alexandria a guardá-los em uma estrutura que se tornou conhecida como biblioteca.

Nos primórdios da computação, trabalhava-se diretamente com as células de memória da mesma forma que os povos helênicos trabalhavam com folhas de papiro. Quando, porém, os dados ficaram muito complexos, decidiu-se organizar as células de memória em estruturas de dados.

As operações com estruturas podem ser divididas em duas etapas: construção e seleção. A construção está relacionada com a montagem da estrutura. No caso da escrita, encadernar e erguer o prédio da biblioteca pertencem à fase de construção. A seleção engloba os métodos de recuperar informações armazenadas em estruturas. Digamos que você queira saber algo sobre Alexandria. Você vai à biblioteca, procura um livro no fichário, pede à bibliotecária para localizá-lo, consulta o índice e acha a informação. Tudo isto é seleção.

Em computação existem dezenas de tipos de estruturas. Alguns deles são: matrizes, listas, cadeias (strings), árvores, pilhas, esquemas, registros, grafos, redes etc. Destas estruturas, BASIC oferece duas: matrizes e cadeias. FORTH é muito mais flexível, pois possui comandos que permitem a criação de qualquer tipo de estrutura concebida pelo programador. Estes comandos são **<BUILDS** e **DOES>**.

ESTRUTURAS EM FORTH

Vamos mostrar como criar tipos de estruturas através de um exemplo. A palavra abaixo definida constrói estruturas do tipo vetor:

```
: VEC < BUILDS 2 * ALLOT DOES > SWAP 1- 2 * + ;
```

Suponhamos agora que você queira criar um vetor com 4 elementos. Basta teclar **4 VEC D**. Vejamos o que aconteceu. Os comandos entre o **<BUILDS e o DOES>** são executados na fase de construção da estrutura. No caso que estamos analisando, a estrutura criada recebeu o nome de **D** e no momento em que foi montada, havia um 4 na pilha. Este foi multiplicado por 2, produzindo 8. **ALLOT** reserva uma quantidade de células de memória dada pelo número que está no topo da pilha, ou seja, 8. Após a execução de **ALLOT**, está terminada a fase de construção. Passemos à fase de seleção. Para colocar o número 15 no segundo elemento de **D**, é mister digitar **15 2 D !**.

Quando **D** é usada, as palavras entre o **DOES>** e o ponto e vírgula são executadas. Inicialmente, **DOES>** coloca na pilha o endereço da primeira célula da estrutura. Se este endereço for 2001, a pilha se tornará:

2001
2
15

A execução de { SWAP 1-2 * + } produzirá o seguinte resultado:

2003
15

Finalmente, o ponto de exclamação enviará o 15 ao endereço 2003.

Para imprimir o conteúdo do segundo elemento de **D**, é suficiente digitar **2 D @** . De novo, as palavras entre **DOES>** e o ponto e vírgula são ativadas e produzem o endereço 2003, do qual **@** retirará o conteúdo 15.

Sempre que for necessário produzir vetores de inteiros, posso usar **VEC**. Por exemplo:

```
7 VEC S
12 VEC M
```

Tanto **S** quanto **M** terão o mesmo comportamento que **D**.

FLEXIBILIDADE

Programadores sem imaginação costumam usar **<BUILDS DOES>** apenas para criar matrizes, cadeias e outras estruturas existentes em BASIC. Sugiro-lhe que não se deixe limitar pela experiência que teve com linguagens convencionais. Alie sua inventividade ao poder de FORTH, a fim de produzir estruturas interessantes e práticas. Veja, por exemplo, uma estrutura para armazenar notas musicais. Vamos chamá-la de **NOTA**.

```
: NOTA < BUILDS  ,  DOES > @ SWAP  NOTE  ;
```

Definamos, pois, algumas estruturas do tipo **NOTA**.

```
167 NOTA DÓH
157 NOTA RÉ
148 NOTA MI
```

No momento da construção, a vírgula que aparece após o **<BUILDS** reserva uma célula de memória para armazenar o topo da pilha. No caso do **RÉ**, é armazenada a freqüência 157. Para usar as novas estruturas, digita-se uma duração seguida do nome da estrutura. Assim:

```
128 RÉ
130 DÓH
300 MI
```

Isto causará a execução do que está entre o **DOES>** e o ponto e vírgula. Concentremo-nos no **MI. DOES>** colocará no topo da pilha o endereço da primeira célula da estrutura e **@** substituirá o referido endereço por 148. Antes da execução de **SWAP**, a pilha ficará assim:

```
148
300
```

A palavra **NOTE**, definida na primeira lição, produzirá um tom musical.

Uma estrutura que por certo interessará aos aficcionados em jogos é **FIG**. Ela serve para armazenar figuras e é definida abaixo:

```
: FIG < BUILDS  DUP C, 0 DO C, LOOP
      DOES > DUP  C@ 0
          DO  1+ DUP C@ EMIT LOOP DROP ;
```

A seguir, mostramos como criar figuras usando **FIG**:

```
129 184 171 151 176 5 FIG COBRA
```

O número 5 indica a quantidade de caracteres gráficos que serão usados para formar a figura. Os inteiros que precedem o 5 são os códigos dos caracteres gráficos. Como estes códigos variam de máquina para máquina, você deve substitui-los apropriadamente, se seu computador não for TRS-80. A palavra **{C,}** reserva 1 byte e armazena o topo da pilha nele. Analise cuidadosamente as palavras que estão entre o **<BUILDS** e o **DOES>** e convença-se de que elas colocarão na estrutura de dados não só o número 5 como também todos os caracteres gráficos.

Quando teclarmos **COBRA**, os caracteres gráficos serão impressos pelos comandos que estão entre o **DOES>** e o ponto e vírgula, produzindo uma figura. E é claro que podemos usar **FIG** para montar outras figuras. Por exemplo:

```
179 179 179 179 179 179 6 FIG TUBO
```

Para finalizar, gostaria de avisar que, em FORTH padrão, o **<BUILDS** se chama **CREATE**. Isto significa que, se você está usando FORTH padrão, a definição de **VEC** se torna:

```
: VEC CREATE  2 * ALLOT DOES > SWAP  1- 2 * + ;
```

Antonio Eduardo Costa Pereira é formado em Engenharia Eletrônica pela Escola Politécnica da USP. Fez mestrado em Ciências Espaciais no Instituto de Pesquisas Espaciais, em S. J. dos Campos, e doutorado em Engenharia Elétrica na Cornell University, de Nova Iorque.

MS É FEITA PARA VOCÊ PARTICIPAR COM SUA OPINIÃO

Escreva-nos dizendo qual a sua área de interesse, conte-nos também as suas experiências com seu micro, o que você quer ver publicado em MS, o que você acha da sua MS, isto irá nos ajudar a fazer de MICRO SISTEMAS a SUA revista.
E lembre-se: todo leitor que nos escreve concorre automaticamente a uma assinatura de um ano de nossa MICRO SISTEMAS.
Mande logo sua opinião para Redação de MICRO SISTEMAS no Rio de Janeiro ou em São Paulo.

Micro Sistemas

Av. Presidente Wilson, 165
grupo 1210 - Centro - CEP 20030
Rio de Janeiro - Tel.: (021) 262-6306

Rua Oliveira Dias, 153
Jardim Paulista
São Paulo/SP - Tel.: (011) 853-7758

Peek & Poke

SOFTWARE PARA CP 400 E COMPATÍVEIS

APLICATIVOS: Processador de Textos PPTEXTO c/ acentuação em português, Planilhas Eletrônicas, Bancos de Dados, Pacotes Integrados.
UTILITÁRIOS: Editor/Assembler, Expansor de Tela, Supertec, Compiladores, Copiadores, Etc.
GRÁFICOS: DUMP P-500, Esquemas Eletrônicos, Bjork Blocks, Comerciais, Graphicom e outros.
LINGUAGENS: LOGO em português, PILOT para professores e uso didático, PASCAL, CBasic.
JOGOS: Muitos títulos para mencionar aqui! Nossos catálogos mostram as telas dos jogos. Temos jogos exclusivos. Escolha melhor na Peek & Poke!

SOFTWARE PARA CP 500 / CP 300

— Processador de textos PPTEXT 500 DISCO c/ acentuação.
— Planilha Eletrônica CALC-300. Única em Z-80 para versão K-7.
— Utilitários copiadores para fita e disco.
— Editor/Assembler, monitor MACMON e tutor de linguagem de máquina. Em fita ou disco. Aprenda Z-80!
— Compilador ZBasic. Produz código objeto com rapidez.

SOFTWARE PARA LINHA MSX

Revendemos cartuchos e fitas c/ os melhores preços. Desenvolvemos títulos próprios. Entre em contato.

A PEEK & POKE demonstra, garante e dá suporte permanente a seus programas. Venha conhecê-los ou solicite catálogo completo para o seu equipamento.

PEEK & POKE Microcomputação e Comércio Ltda.
Av. Brig. Faria Lima, 1664 - Conj. 1102
01452 - São Paulo - SP - Fone: (011) 813-3277

"MIKROS"

- Microcomputadores Pessoais e Profissionais, Software, Suprimentos e Cursos.
- Financiamento em até 18 meses sem entrada e os preços mais baixos do mercado.
- Atendimento perfeito, profissionais treinados e habilitados para dar a você a certeza de um bom Investimento.

EQUIPAMENTOS
Micros das linhas: TRS-80 • Apple • IBM • Sinclair • TRS-80 Color
Impressoras • Vídeos • Interfaces • Etc.

SUPRIMENTOS
Formulário Contínuo • Disquetes • Fitas • Mesas • Etiquetas • Etc.

SOFTWARE
Nacionais e Importados mais de 2.000 programas e jogos de todas as linhas.

CURSOS
Basic • Basic Avançado e DOS.

Av. Ataufo de Paiva, 566
sobreloja 211 e 202
Rio de Janeiro - R.J.
Tels.: (021) 239-2798 e 511-0599

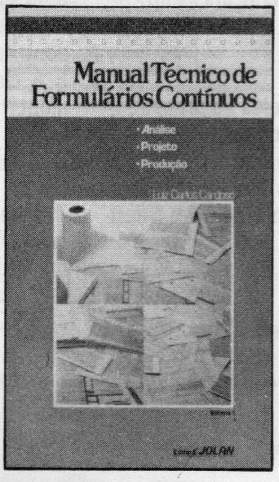
Livros

PFUHL, D. M. A. V.; TUCCI, W. J., *LOGO – Programação e Aprendizado*, Editora Nobel.

LOGO – Programação e Aprendizado destina-se a pessoas de todas as idades, inclusive aquelas que nunca utilizaram um computador. Nos três capítulos iniciais, os autores apresentam a linguagem LOGO, descrevendo seus primeiros passos com detalhes. Posteriormente, eles transmitem os conceitos básicos de programação através de exercícios práticos sempre acompanhados de exemplos e ilustrações.

ABREU, C. A. C. de, *47 programas para ZX Spectrum e TK90X*, Edições Microkit.

O livro traz 47 listagens para a linha ZX Spectrum que dão ao usuário a chance de exercitar progressivamente a programação em seu micro. Entre os programas listados estão: Relógio Digital; Controle Bancário; Leitor de Código de Máquina; Simulação da Loto; Órgão; Medidas Estatísticas; Minieditor de Textos; Gráfico de Setores; Inversão de Matrizes e Jogo de Dados.



CARDOSO, L. C., *Manual Técnico de Formulários Contínuos*, Editora Jolan.

Este lançamento da editora Jolan aborda um assunto até então pouco explorado nos livros especializados: os formulários contínuos. O livro apresenta mais de duzentas ilustrações, fotos e tabelas e trata de assuntos como as características dos formulários; seus elementos básicos; a matéria-prima utilizada; a produção; formulários comerciais; técnicas de impressão contra falsificação e a terminologia gráfica dos formulários contínuos.

MIRSHAWKA, V., *Conexão Grafix-IBM-PC/XT e compatíveis*, Editora Nobel.

Este livro descreve os recursos da conexão entre as impressoras Grafix e os diversos micros compatíveis com o IBM-PC/XT, como por exemplo a possibilidade de se acionar oito agulhas ao mesmo tempo quando se está no modo gráfico.

Os 13 capítulos do livro foram escritos com a colaboração de vários fabricantes de micros de 16 bits, que apresentaram programas-exemplos capazes de destacar as qualidades de cada equipamento.

PEREIRA, C., ALCANTARA, R. B., *Enciclopédia da Linguagem BASIC*, Editora Campus.

A *Enciclopédia da Linguagem BASIC* apresenta, em 496 páginas, instruções, comandos e funções para a programação BASIC em equipamentos de diversas linhas. Eles são acompanhados de exemplos que visam a esclarecer dúvidas, além de pequenos programas para testar os subcomandos e subrotinas que facilitam o uso dos programas.

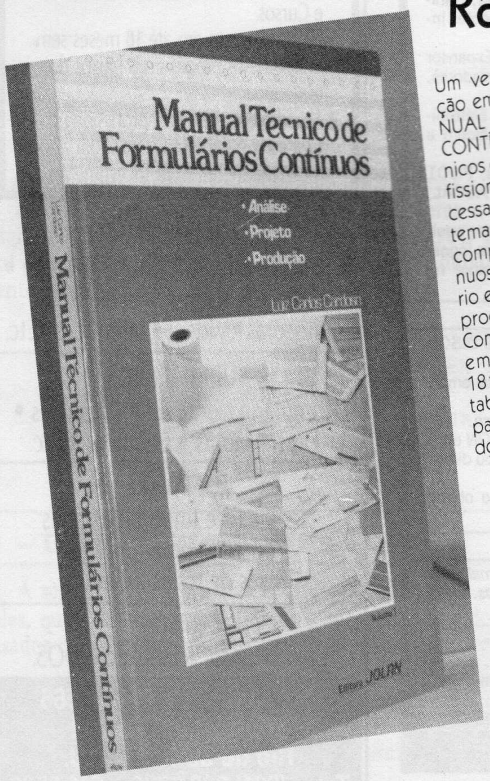
Na introdução da obra, o autor trata brevemente de assuntos introdutórios. Os comandos, instruções e funções, que vêm logo a seguir, são apresentados em ordem alfabética, e cada um deles traz definição e exemplos, além de ortografias alternativas, variações de uso etc.

ENDEREÇO DAS EDITORAS

Editora Campus – Rua Barão de Itapagipe, 55, CEP 20261, tel.: (021) 284-8443, Rio de Janeiro.
Editora Jolan – Av. Rio Branco, 156, sala 1537, CEP 20040, tel.: (021) 262-4619, Rio de Janeiro.
Editora Nobel – Rua da Balsa, 559, CEP 02910, tel.: (011) 257-2144, São Paulo.
Edições Microkit – Rua Visconde de Pirajá, 303, grupo 1005, CEP 22410, tel.: (021) 521-4638, Rio de Janeiro.

Economize tempo e dinheiro.

Racionalize seu 'business forms'



Um verdadeiro curso de especialização em "business forms", o livro MANUAL TÉCNICO DE FORMULÁRIOS CONTÍNUOS é indispensável para técnicos e estudantes de informática, profissionais de administração, O&M, processamento de dados, analistas de sistemas, programadores, vendedores e compradores de Formulários Contínuos. Ensina como projetar o formulário em função do seu equipamento de processamento.
Com aproximadamente 248 páginas em papel off-set, no formato 18x27cm, todo ilustrado com fotos e tabelas, capas impressas a 4 cores em papel especial e acabamento costurado e encadernado.

preço por exemplar
Cr$ 250.000
Validade: até 28/2/86

Desejo receber ( ) exempla(es) do livro Manual Técnico de Formulários Contínuos, pelo(s) qual(is) pagarei a quantia de Cr$ _____ (_____).
Nome _____
Endereço _____
Cidade _____ Estado _____ CEP _____

Para tal estou enviando um cheque à ATI – EDITORA LTDA., com despesas de entrega incluídas.

ATI – EDITORA LTDA.
Av. Pres. Wilson, 165 - Gr. 1210 - Tel.: 262-6306 - CEP 20030
Rio de Janeiro, RJ.

MS SERVIÇOS Serviços Serviços Serv

APPLE SOFT?

Venha para o clube diferente de Apple

"MAGIC WORLD CLUB"

Escreva para nós e tenha uma apple surpresa.

Caixa Postal 62521.
CEP 01150
Tel.: (011)664316
São Paulo-SP

SUPRIMENTO É COISA SÉRIA

- DISKETTES: 5 1/4 e 8" e fitas magnéticas — marca **DATALIFE VERBATIM**
- ETIQUETAS **PIMACO — PIMATAB**
- FORMULÁRIOS CONTÍNUOS E PASTAS
- FITAS PARA IMPRESSORAS EM GERAL
- ARQUIVOS PARA DISKETTES

PRODUTOS COM GARANTIA E ENTREGA IMEDIATA

CENTRALDATA Com. e Representações Ltda.
AV. PRES. VARGAS Nº 482 GR. 201/203
Tel.: KS (021) 253-1120
Telex: (021) 34318

SOFWARE BARATO!

A ALFAMICRO coloca a sua disposição os melhores programas do mercado internacional ao menor preço.

PROGRAMAS PARA APPLE

Escolha os seus entre mais de 2.000 títulos que cobrem as mais variadas aplicações a Cr$ 35.000 por disco.

PROGRAMAS PARA CP-500

Os mais famosos títulos a Cr$ 45.000 por disco.

POSSUIMOS TAMBÉM PROGRAMAS PARA IBM-PC e S-700

Escreva já! E recebe nosso catálogo.

ADQUIRA PELO CORREIO OS MELHORES PREÇOS PERIFERICOS E ACESSORIOS PARA APPLE E IBM-PC.

CONSULTE-NOS. COBRIMOS QUALQUER OFERTA!

ALFAMICRO INFORMÁTICA
Cx. Postal, 12.064 — 02098
F. 011 - 950-8998 - São Paulo - SP

COMPUCLUB

Um CLUBE MUITO ESPECIAL Para usuários dos equipamentos TK-85, TK90X, CP-400, CP-500 e compatíveis

NOVA FASE 1986

VOCÊ RECEBE, INTEIRAMENTE GRÁTIS:

— Um curso completo de BÁSIC.
— Edições Mensais do Compuclub News, com programas de jogos, aplicativos e dicas especiais para o seu equipamento.
— A cada 45 dias, programas amplamente documentados, com seus manuais de instrução, gravados em fita HOT LINE, a melhor opção para o seu acervo de softs.

E AINDA GANHA O DIREITO DE ADQUIRIR, PELO MENOR PREÇO DO MERCADO, MICROS, SUPRIMENTOS E ACESSÓRIOS.

Associando-se agora, você ainda recebe os 5 boletins já editados pelo clube, incluindo o especial dezembro/85.
Solicite, ainda hoje, informações detalhadas acerca de como participar do COMPUCLUB. Não se esqueça, porém, de indicar o tipo de micro que você possui.

COMPUCLUB — Caixa Postal 46 (36570) Viçosa-MG

REG. SEI N.º 0219

SBL DADOS sistemas de banco de dados

CURSOS DISPONÍVEIS

- Introdução à Microcomputação
- DOS - PC "Sistema Operacional"
* - UNIX "Sistema Operacional"
- LINGUAGEM C "Ling. Programação"
- dBASE II "Programação Básica"
- dBASE II "Programação Avançada"
- dBASE III "Programação Básica"
- LOTUS 1-2-3 "Plan. Eletrônica"
* - Framework "Sistema Integrado"
- Symphony "Sistema Integrado"
- Wordstar "Processador de Texto"

* Em desenvolvimento

MATERIAIS DIDÁTICOS: Publicações Técnicas desenvolvidas em Português.
RECURSOS DIDÁTICOS: Conceitos e exemplos práticos, através de Micros e Telão de 72".
CURSOS FECHADOS E ABERTOS

NOVO ENDEREÇO Rua Groelândia, 1.750 - Jardim Europa
CEP 01434 - São Paulo - SP.
Telefones: 881-4599 - 881-4032

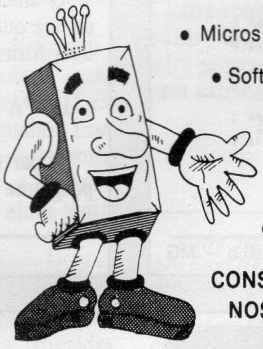
BASIC e COBOL

CURSOS DELTA/DATA RIO

- Turmas às 2as e 4as, 3as e 5as e aos sábados.
- Microcomputadores CP-500 da Prológica.
- Plantão de dúvidas.
- Preços promocionais parcelados.
- Apostilas e certificado de conclusão.
- "Clube do Micro" às 6as onde o CP-500 é todo seu:

Rua Machado Bitencourt, 278 (011) 549-9811 S. Paulo — SP
Rua Voluntários da Pátria, 286 - 2º (021) 286-4740 - R. de Janeiro - RJ

O "PACOTÃO" OFERECE:



- Micros
- Software
- Assistência Técnica IBM/PC, XT e AT
- Suprimentos
- Acessórios Periféricos

CONSULTE NOSSOS PREÇOS!

LIGUE JÁ!
(011) 276.8988

DATAROAD
Rua Luiz Goés, 1894 — São Paulo
CEP 04043 — Telex: (011) 37755 DTRD

dicas

Envie suas dicas para a Redação de MICRO SISTEMAS na Av. Presidente Wilson, 165 - grupo 1210. Centro, Rio de Janeiro, RJ. CEP 20030

Linha SINCLAIR

Título animado

Esta rotina em BASIC cria no seu micro uma interessante apresentação do conteúdo da variável A$ da linha 10, que deverá ter no máximo 32 caracteres. Na linha 9040, você pode controlar a linha final da impressão da sua frase, que, neste exemplo, é 15. Use-a como sub-rotina para incrementar seus programas.

```
  10 LET A$="MICRO SISTEMAS"
  20 GOSUB 9000
  30 STOP
9000 FOR X=1 TO LEN A$
9010 IF A$(X)=" " THEN NEXT X
9020 LET Z=(15-LEN A$/2)+X
9030 GOSUB 9050
9040 FOR A=20 TO 15 STEP -1
9050 PRINT AT A;Z;A$(X);AT A+1,Z
;" "
9060 NEXT A
9070 NEXT X
9080 RETURN
9090 FOR A=30 TO Z STEP -1
9100 PRINT AT 21,A;A$(X);" "
9110 NEXT A
9120 RETURN
```

Carlos Alexandre da Silva — RJ

Linha APPLE

PEEKs e POKEs I

- Retorne o endereço de início das variáveis:
  PEEK(105) + 256 * PEEK(106)
- Retorne o endereço de início das matrizes:
  PEEK(107) + 256 * PEEK(108)
- Retorne o endereço do fim das variáveis numéricas:
  PEEK(109) + 256 * PEEK(110)
- Retorne o endereço de início das Strings:
  PEEK(111) + 256 * PEEK(112)
- Retorne o valor do HIMEM:
  PEEK(115) + 256 * PEEK(116)
- Retorne o endereço do fim do espaço das variáveis:
  PEEK(204) + 256 * PEEK(205)
- Faça a primeira linha ser listada continuamente quando dermos LIST:
  POKE 2049,1.

Esta dica também serve para o TK 2000.

PEEKs e POKEs II

- Retorne o valor de Maxfiles:
  PEEK(43607)
- Retorne o último número atribuído ao parâmetro D:
  PEEK(43624)
- Retorne o último número atribuído ao parâmetro S:
  PEEK(43626)
- Retorne o volume do disco:
  PEEK(46017)

Obs.: Estes valores se aplicam a sistemas com 48 Kb. Para os de 32 Kb, subtraia 16384 do valor. Para sistemas com 16 Kb, subtraia 32768 do valor.

Bruno Cabral — PB.

Linha ZX Spectrum

Túnel e pirâmide

Veja como o comando DRAW pode ser útil na criação de figuras geométricas, em seu compatível com o ZX Spectrum:

```
 5 INK 0: PAPER 6: BORDER 1
10 FOR E=0 TO 127
20 PLOT E,E: DRAW 255-E*2,0
30 NEXT E
```

```
 5 INK 0: PAPER 6: BORDER 1
10 FOR F=80 TO 0 STEP -2
20 PLOT F,F: DRAW 255-F*2,0
30 DRAW 0,175-F*2
40 DRAW -255+F*2,0
50 DRAW 0,-175+F*2
60 NEXT F
```

Teotônio Rodrigo de Carvalho — MG

Linha TRS-COLOR

PAUSA

Freqüentemente é necessário interromper a execução de um programa até que o usuário esteja pronto para continuar. Um método empregado para conseguir isto é o que se segue:

5000 PRINT "APERTE QUALQUER TECLA PARA CONTINUAR"
5010 A$ = INKEY$: IF A$ = " " THEN 5010

O único problema é ter que ficar repetindo o **IF INKEY$ = " " THEN**... toda vez que uma pausa for necessária. Os usuários do Color podem fazer isso de uma forma mais simples: é só incluir no começo do programa uma linha do tipo:

10 PA = 44539 (ou outra variável qualquer)

A partir daí, basta usar **EXEC PA** toda vez que for necessário interromper o programa até que alguma tecla seja pressionada.

Cláudio Costa — RJ

Linha TRS-80

Trave o BREAK

Dê um POKE 16396,165. A tecla BREAK não **BREAKará** o programa. Ele fica inerte, sem efeito nenhum. Com isto, no DOS, se você gravar seu programa incluindo o POKE logo no começo, quando o *pirata* der o RUN "nome/sigla" e tentar dar o BREAK, dançará ao som de nossa música.

Bruno Lopes Fernando Cabral — PB

Linha APPLE

Palavras em movimento

É muito simples, usando os pequenos programas abaixo você irá causar um bonito efeito em seu vídeo. Porém há um inconveniente, a frase (que no caso está sendo representada pela string A$) não pode ter mais de 40 (quarenta) caracteres, o que equivale a uma linha.

As duas linhas abaixo serão comuns a todos os programas, ou seja, todos têm as linhas 10 e 20 iguais:

```
10 HOME
20 A$=" * M.S. A PRIMEIRA EM MIC
ROCOMPUTADOR * "
```

A. Da direita para a esquerda:

```
30 FOR X=1 TO 40
40 VTAB 1:HTAB 41-X:PRINT LEFT$(
A$,X)
50 FOR R=1 TO 75
60 NEXT R,X
70 END
```

B. Da esquerda para a direita:

```
30 FOR X=1 TO 40
40 VTAB 1:PRINT RIGHT$(A$,X)
50 FOR R=1 TO 75
60 NEXT R,X
70 END
```

C. De cima para baixo:

```
30 FOR X=2 TO 23
40 VTAB X-1:CALL-868
50 VTAB X:PRINT A$
60 FOR R=1 TO 75
70 NEXT R,X
80 END
```

D. De baixo para cima:

```
30 FOR X=23 TO 1 STEP-1
40 VTAB X+1:CALL-868
50 VTAB X:PRINT A$
60 FOR R=1 TO 75
70 NEXT R,X
80 END
```

Ricardo F. de Miranda – SP

Linha ZX Spectrum

Raios coloridos

O DRAW é um recurso gráfico com efeito, principalmente quando é usado com um ponto de partida (PLOT, CIRCLE). Para provar isso, aí vai uma dica quente:

```
   1 REM   Micro Sistemas
Carlos Henrique Olifiers-12 anos
  10 INPUT "Qual e' o numero de
DRAWS ? ";A
  20 CLS
  30 FOR B=1 TO A
  40 INK RND*5
  50 LET C=INT (RND*70)+2
  60 LET D=INT (RND*80)+1
  70 PLOT C,D
  80 DRAW C,D
  90 NEXT B
 100 GOTO 10
```

Carlos Henrique Olifiers – RJ

Linha TRS-COLOR

INKEY$

A maneira mais comum de simular movimento em jogos animados escritos em BASIC (quando, por qualquer motivo, não se deseja usar o joystick) é utilizar INKEY$ em combinação com as teclas das setas. As seguintes instruções movem um bloco amarelo para a esquerda, quando se pressiona a tecla (CHR$ 8); e para a direita, se for pressionada a tecla (CHR$ 9):

```
5 CLS 0 : X$ = CHR$ (128) + CHR$ (159) + CHR$ (128) : N = 400
7 PRINT @ N,X$;
9 I$ = INKEY$ : N = N + (I$ = CHR$(8) AND N > 383) – (I$ = CHR$
  (9) AND N < 414) : GOTO 7
```

A desvantagem desse método, além de ser muito lento, é que você tem que largar a tecla e pressioná-la novamente toda vez que quiser mover o bloco uma posição. Para resolver os dois problemas de uma só vez, dê uma olhada na tabela abaixo: ela fornece as posições da memória correspondentes às teclas das setas que, quando pressionadas, assumem o valor 247. Agora, é só substituir INKEY$ pelos valores das teclas. Rode o exemplo a seguir e veja a diferença:

```
9 N = N + (PEEK(343) = 247 AND N > 383) – (PEEK(344) = 247 AND
  N < 414) : GOTO 7
```

TECLA	POSIÇÃO
↑	341
↓	342
←	343
→	344

Cláudio Costa – RJ

Linha ZX81

Explosão

A rotina abaixo apresentada se propõe a criar uma explosão na tela. Para digitá-la, crie uma linha (REM) com 22 caracteres e para usá-la, basta dar o (RAND USR) 16514. Uma boa dica para ser utilizada em jogos.

```
16514  06 0A 0E 00 16 1E 7A ED
16522  47 1E 20 FA 0D 20 F5 10
16530  F1 3E 1E ED 47 C9
```

Aloisio Carlos Tortelly Costa Jr. – RJ

Linha TRS-80

Desenhos

Desenhe círculos, elipses e outros efeitos com esta dica. Para criar novas figuras, mude os parâmetros da linha 40.

```
5 CLS
20 FORF=0TO2*3.1416STEP3.1416/18
30 FORN=1.5TO20
40 SET(60+2*N*SIN(F),20+N*COS(F))
60 NEXTN
70 NEXTF
80 GOTO5
```

João Luiz Pacheco Vieira Galvão – GO

Linha SINCLAIR

Conversão decimal/binário

O TK90X possui uma função (BIN), que permite se obter um número inteiro a partir do seu correspondente binário.

Use esta dica para conseguir o inverso, ou seja, tendo o número inteiro, conseguir o seu binário correspondente:

Para saber se o bit B do número N é 0 ou 1, faça:

```
LET L=INT (N/2**B)-2*INT ((INT (N/2**B))/2)
depois use PRINT L, para saber o valor do bit.
```

Para ver todos os bits de um número N de 2 bytes, rode o seguinte programa:

```
  10 INPUT N
  20 PRINT N;TAB 10;
  30 FOR B=15 TO 0 STEP -1
  40 PRINT INT (N/2**B)-2*INT ((
INT (N/2**B))/2);
  50 IF B=8 THEN PRINT " ";
  60 NEXT B
```

Para rodar esta dica nos compatíveis com ZX Spectrum, substitua os sinais ** pelo sinal de potenciação(↑).

Cláudio Bittencourt – SP.

A criação da linha MSX representa uma verdadeira revolução no mundo da informática. Leia este artigo e fique por dentro das características desses micros, seus recursos e periféricos.

O padrão MSX

Oscar Júlio Burd e Luiz Sérgio Y. Moreira

Uma das grandes atrações da V Feira Internacional de Informática foi a apresentação de microcomputadores padrão MSX por dois fabricantes nacionais: Gradiente, com o Expert, e Sharp, com o Hot Bit.

Este é o primeiro de uma série de artigos que têm por objetivo mostrar desde o nascimento até as principais características e inovações atuais dos micros da família MSX (Microsoft Super extented), incluindo desde o poderoso BASIC, da Microsoft, até os jogos, utilitários e periféricos existentes no Brasil e no exterior.

O NASCIMENTO DE UM PADRÃO

Os primeiros microcomputadores foram criados por entusiastas da eletrônica em garagens e fundo de quintal. Naturalmente, estes pioneiros não tinham a mínima preocupação a respeito de como os circuitos deveriam ser configurados e de como o software deveria trabalhar.

Este fato deu origem à grande variedade de micros, hoje existentes, sendo que, na maioria dos casos, os programas que funcionam em uns não podem ser utilizados em outros (por exemplo, programas feitos para o Apple não funcionam em micros TRS-80 e outros não compatíveis).

Tanto os fabricantes de micros como as software-houses e o consumidor saem prejudicados pela falta de um padrão único: as software-houses têm de fazer várias versões de um mesmo programa, uma para cada modelo, acabando por encarecer o preço de cada versão; os fabricantes correm o risco de fechar as portas se poucas software-houses dedicarem-se a produzir programas para seus micros. Quanto ao consúmidor, para ele é repassado o custo final, sempre lembrando que o consumidor fica restrito a usar apenas os programas para a linha a qual seu micro pertence.

EMPRESAS QUE ADERIRAM AO PADRÃO MSX	
CANON	JAPÃO
CASIO KEISANKI	JAPÃO
FUJITSU	JAPÃO
GENERAL	JAPÃO
HITACHI SEISAKUSYO	JAPÃO
KYOCERA	JAPÃO
MATSUSHITA DENKI (NATIONAL)	JAPÃO
MITSUBISHI DENKI	JAPÃO
NIHON DENKI (NEC)	JAPÃO
NIHON GAKKISEIZOU (YAMAHA)	JAPÃO
NIHON VICTOR (JVC)	JAPÃO
PHILIPS	HOLANDA
PIONEER	JAPÃO
SANYO	JAPÃO
SHARP	JAPÃO
SONY	JAPÃO
SPECTRAVIDEO	USA
TOSHIBA	JAPÃO
MITSUMI DENKI	JAPÃO
DAEWOO	COREA DO SUL
GOLDSTAR	COREA DO SUL
GRADIENTE	BRASIL
SHARP DO BRASIL	BRASIL

Figura 1

```
CARACTERÍSTICAS BÁSICAS DO MSX

UCP: Z80 a 3,58 MHZ ou compatível
ROM: 32 Kb
RAM: 8 Kb (mínimo)
RAM VÍDEO (VRAM):  16 Kb
TEXTO: 24 linhas x 40 colunas (modo 0)
       24 linhas x 32 colunas (modo 1)
GRÁFICO: 256 x 192 pontos (modo 2)
         64 x 48 pontos (modo 3)
COR: 16 cores
CONTROLE DE VÍDEO: TMS-9128NL ou
                   compatível
GERADOR DE SOM: AY-3-8910 ou compatível
PPI: INTEL 8255
INTERFACE CASSETE: 1200/2400 bauds FSK
BARRAMENTO: 50 pinos
IMPRESSORA: paralela, norma Centronics
CONECTORES DE JOYSTICK: 1 (mínimo) ou
                        2 (máximo)
INTERFACE DE ÁUDIO: 1 conector
```

Figura 2

Tendo em vista as desvantagens da não padronização, duas empresas, a Microsoft (americana) e a ASC II (japonesa), uniram-se em *joint-venture* com o objetivo de criar um padrão para micros que viesse a ser adotado e vendido em larga escala.

As duas empresas conseguiram o que parecia impossível: a criação de um padrão de hardware/software para micros de 8 bits em cooperação com grandes nomes da indústria japonesa: o padrão MSX. A ordem cronológica de adesão das empresas ao padrão consta na figura 1.

Para um computador pertencer à família MSX, ele deve respeitar as seguintes características básicas (veja também a figura 2):

Microprocessadores: o microprocessador deve ser o famoso Z80A (ou compatível) de 8 bits, acompanhado de dois outros microprocessadores: um para som (AY-3-8910 ou compatível) e outro para vídeo (TMS-9128-NL ou compatível). O AY-3-8910 é o responsável pelos dons musicais do MSX, com as seguintes características: três canais de som, um canal de ruído para efeitos

especiais, oito oitavas, oito envoltórias e ajuste do volume em 16 níveis.

Já o TMS-9128-NL fornece quatro tipos de telas, com 16 cores cada: duas telas de texto (40 x 24 e 30 x 24 pontos) e duas telas gráficas (256 x 192 e 64 x 48 pontos), incluindo o uso de *sprites* em 32 planos (*sprites* são figuras coloridas que o usuário cria e manipula em vários planos).

Linguagens: a linguagem de programação deve ser o BASIC da Microsoft. Este BASIC ocupa 32 Kb de ROM e será discutido por nós, em detalhes, futuramente.

Disco Flexível: a formatação do disco deve ser compatível com o sistema operacional adotado que é o MS-DOS da Microsoft. Este sistema operacional é utilizado na maior parte dos micros de 16 bits, como o IBM-PC, por exemplo.

Joystick: no mínimo um e no máximo dois.

Impressora: deve ser paralela, de padrão Centronics.

Gravador Cassete: a transmissão de dados deve ocorrer a 1200 ou 2400 bauds, sendo realizada no padrão FSK. Também deve existir controle sobre o motor do gravador (liga/desliga) a partir do micro.

Slot: deve haver pelo menos um slot externo para conexões com periféricos (unidade de disco flexível, cartuchos com programas etc.).

Saída de Áudio: um conector.

Note-se que os micros do padrão MSX possuem recursos gráficos e sonoros de alto nível, sendo também os mais rápidos dos micros de 8 bits (devido aos três microprocessadores, os MSX podem realizar simultaneamente tarefas como tocar música, fazer animação gráfica e ainda ter a UCP livre para, por exemplo, realizar cálculos e executar programas).

PROCESSADOR DE VÍDEO

O processador de vídeo, também chamado de VDP (*Video Display Processor*), é um dos pontos fortes do MSX, responsável pela simplicidade com que se criam e manipulam imagens coloridas.

Graças ao VDP, o usuário pode criar até 256 figuras de 8 x 8 pontos ou até 64 figuras de 16 x 16 pontos. Essas figuras são chamadas de *sprites*, os quais podem ser colocados em 32 planos diferentes (com um total de 16 cores a escolher).

cód.	cor	cód.	cor
0	transparente	8	vermelho
1	negro	9	vermelho claro
2	verde	10	amarelo escuro
3	verde claro	11	amarelo claro
4	azul escuro	12	verde escuro
5	azul claro	13	magenta
6	vermelho escuro	14	cinza
7	azul celeste	15	branco

Figura 3

É possível, além disso, movimentar os *sprites*, criando um efeito especial de sobreposição, ou seja, os *sprites* dos planos posteriores (mais fundos) passam por trás dos *sprites* dos primeiros planos.

Criando-se um *sprite*, é permitida sua colocação simultânea em vários planos, cada qual com uma coordenada própria. Complementando, ao ocorrer a colisão de dois *sprites*, o VDP gera uma interrupção que pode ser detectada com o uso de um comando do BASIC. Tais recursos são importantes na criação de jogos e animação.

Outras facilidades são o uso de *sprites* numa tela de texto, a possibilidade do uso do comando PRINT para se escrever em tela gráfica com as 16 cores existentes (veja na figura 3) etc.

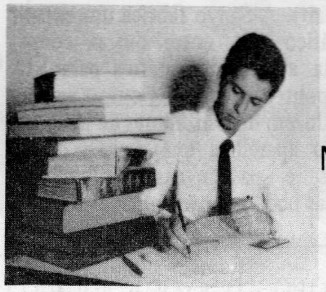
Em termos de comparação, o Apple, que é o micro que mais se aproxima dos MSX, oferece apenas seis cores em alta resolução, não possui *sprites* nem planos de imagem e nele a criação de figuras (shapes) é muito mais trabalhosa. Na figura 4 você pode ver um jogo que se utiliza do recurso dos *sprites*.

O PROCESSADOR DE SOM

O processador de som, também conhecido por PSG (*Programmable Sound* »

"BOM, É QUE NÃO CAI NA MONOTONIA. AQUI MESMO NA EMPRESA EU FAÇO CURSOS DE APRIMORAMENTO"

Fernando Lopes
Técnico de Manutenção



"O PLIM PLIM DA TV PARECIA O BIP. AÍ EU SAÍA CORRENDO"

Rogério de Figueiredo
Técnico Pleno

"MINHA MULHER NÃO ACORDA MAIS COM O BIP. EU VOU E VOLTO, E ELA NEM PERCEBE"

Luis Alberto Silva
Técnico de Manutenção

S H Publicidade

"NUNCA DEIXAMOS UMA MÁQUINA PARADA. SEJA QUAL FOR, ONDE FOR, VAMOS CONSERTAR!"

Mauro Diniz
Auxiliar Técnico

CONSERTO E MANUTENÇÃO DE COMPUTADORES

021 240-4934
011 240-5899

MICROSERVICE

SUPORTE DE INFORMÁTICA

Generator), é outro importante componente do *espetáculo MSX*, capacitando o usuário a criar efeitos sonoros e músicas incríveis através de seus três canais independentes de som (e mais um para ruído).

São tantas as possibilidades que o PSG oferece que vamos tratá-la por etapas:

Volume: o volume de cada canal pode ser controlado independentemente, através do BASIC, em 16 intensidades diferentes. Isto permite que um canal toque uma música, enquanto é acompanhado ao fundo pelos outros dois canais. Como você faria para realizar esta proeza nos outros micros existentes no mercado?

Formas de onda: além de controlar o volume, você pode fixar o formato das ondas sonoras, conseguindo assim sons de instrumentos como piano, órgão, flauta andina, celesta, banjo etc.

Ao todo existem fez formatos de ondas diferentes, todos acessados pelo comando PLAY do BASIC (observe a figura 5).

Canal de ruído: também usado para incrementar os efeitos sonoros, este canal gera ruído branco (algo parecido com o chiado de uma TV fora-do-ar) com freqüência controlável. Este som pode ser mixado independentemente, com qualquer dos três canais.

Com o auxílio do canal de ruídos, pode-se gerar sons tais como os de helicópteros, explosões, turbinas, flauta andina etc. O BASIC permite ainda que se controle o período de execução da música (em notas/minuto), a duração de cada nota, o controle de pausas e que se especifique as notas musicais num total de oito oitavas.

Lembre-se que ao mesmo tempo que você realiza uma animação com o VPD, o PSG pode estar executando uma música, e o Z80 realizando operações variadas!

PERIFÉRICOS

Desde que o fabricante respeite as características básicas do padrão MSX, ele também pode acrescentar elementos adicionais ao seu micro.

O número de periféricos existentes para o MSX é enorme; entre eles estão: teclados de órgão eletrônico, caneta ótica, unidades de disco flexível de 3 1/2", 5 1/4" e 8", monitores monocromáticos e coloridos, mouse, interface para toca discos a laser, interface para vídeo cassete, cartuchos de memória ROM com jogos e utilitários, plotter, cartão de 80 colunas, leitora de código de barras etc.

No exterior, várias indústrias que fabricam o MSX também produzem

Figura 4

algum tipo de periférico. Por exemplo, a Yamaha, que é conhecida mundialmente por sua atuação na área de instrumentos musicais, fabrica teclados e periféricos especiais para a elaboração de sons (contando inclusive com chips adicionais). A Sanyo fabrica um expansor gráfico que, entre outras coisas, aumenta a resolução gráfica para 512 x 204 pontos. Já a JVC e outras empresas produzem digitalizadores de imagem, que são aparelhos que digitalizam as imagens e passam-nas aos micros, onde elas podem ser trabalhadas.

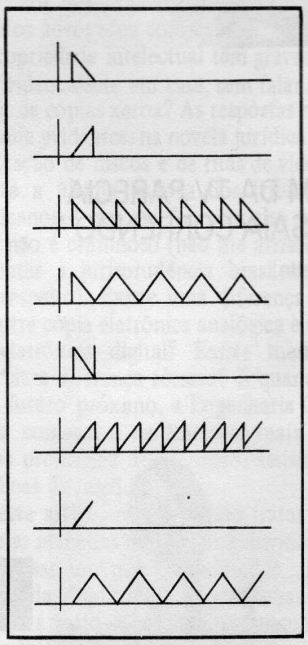
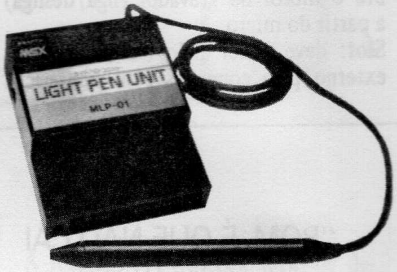


Figura 5 – Forma do envelope.



Light pen, mais um periférico disponível para a linha MSX.

Cabe ressaltar que inúmeras empresas fabricam robôs que podem ser conectados aos MSX e são vendidos como brinquedos (!), no Japão, a preços acessíveis.

Um periférico comum em todos os lares é o amplificador de som que também pode ser conectado diretamente aos MSX, permitindo maior pureza e qualidade de som.

Muito popular no exterior, o modem permite a conexão dos micros entre si ou com mainframes, possibilitando a comunicação com grandes bancos de dados, emulação de terminais ou simples transferências de documentos e programas.

Outro grande trunfo desta linha de equipamentos é que, embora os periféricos sejam fabricados por muitas empresas, todos funcionam em qualquer micro do padrão MSX. Quem sai ganhando com isto é o consumidor, pois tem a sua disposição todo o universo de periféricos do mercado.

Por questão de espaço, descrevemos

apenas alguns dos periféricos existentes, deixando porém o alerta de que há muitos deles à disposição, e que logo chegarão ao nosso país. Por hora, os fabricantes nacionais já prometem o lançamento de data-recorder (gravador específico para uso em informática), joysticks e modem.

OS JOGOS

O uso mais popular dos MSX, além de programas educacionais, está nos jogos. Existem dezenas (se não centenas) de games para esta linha, sendo a maior parte em cartuchos (tendência que, parece, será seguida aqui no Brasil). Essa pluralidade de programas traz também uma grande variedade nos tipos de jogos, mas a maioria pode ser classificada em dois grupos: os de aventura (*adventures*) e os do tipo fliperama (*arcade*).

Os jogos mais famosos que estão no mercado de videogames foram adaptados para os MSX, com muitas melhorias, tanto na parte visual (maior número de detalhes) quanto no que se refere ao som. Além disso, existem novos jogos, principalmente no Japão, sem similares no mundo inteiro. Este é o caso dos programas feitos nos MSX que operam

As mais modernas tecnologias, como o disco a laser, podem ser conectadas ao padrão MSX.

em conjunto com um videolaser, permitindo um grau de detalhes e animação jamais vistos até então.

Aqui no Brasil, o usuário também não terá queixas dos jogos para o seu MSX: ele também vai poder contar com muitos detalhes gráficos (de formas e cor), complementados pelos efeitos sonoros dignos da família.

Estes são alguns dos jogos que já podem ser encontrados:

Pastfinder-galax – aventura em um deserto radioativo, entre ruínas e armadilhas de uma guerra nuclear. O jogador tem que destruir discos voadores inimigos e capturar cargas, sempre tomando cuidado com o nível de radiação que é variável. Vem em cartucho.

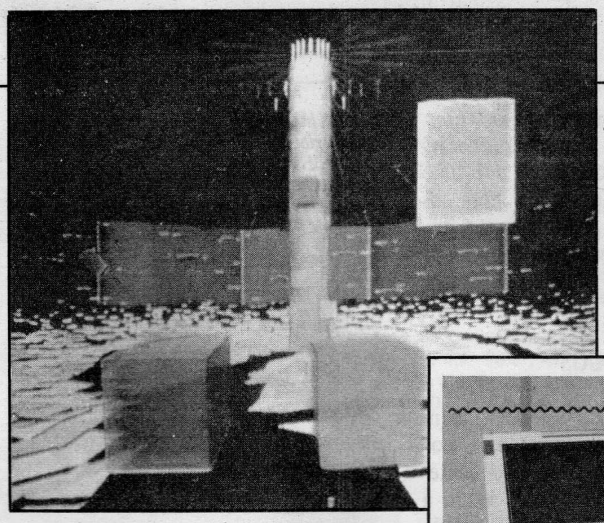
Estrela polar – o objetivo deste jogo é destruir uma nave mãe com um míssil especial, enfrentando esquadrilhas de naves inimigas, as quais atacam inclusive »

pelas costas. Para ajudar o piloto, existe um radar a bordo. Vem em fita cassete.

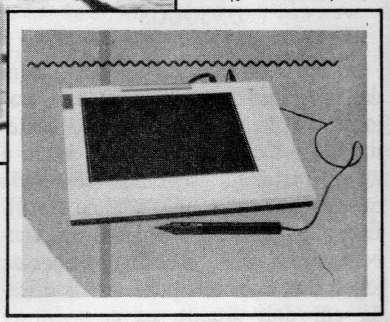
Columbia – o jogador pilota a astronave Columbia através de esquadrilhas suicidas e armadilhas voadoras, tentando conquistar um território desconhecido. Existe um jogo similar para micros da linha Apple, mas inferior, tanto a nível visual quanto sonoro. Vem em cartucho.

Blast-off! – a missão do jogador é destruir todas as naves e esquadrilhas inimigas da sua região cósmica. Em dado instante, o MSX fala Blast-off! Vem em cartucho.

Rollerball – jogo tipo fliperama. O jogador pode comandar três pares de flippers em três campos de jogos diferentes, simultaneamente. Vem também em cartucho.



Um novo padrão de games (foto ao lado) foi inaugurado com o uso dos discos a laser. A mesa gráfica, instrumento indispensável em CAD/CAM, é outro periférico existente para os micros MSX (foto abaixo).



Lode runner – o objetivo é escapar de labirinto gigantesco, tendo que livrar-se ainda de terríveis inimigos, para tanto dinamitando os caminhos deles. O jogo tem 76 níveis e vem em cartucho.

Os MSX nacionais

Aportaram, recentemente, em nossas areias, os primeiros MSX nacionais, por intermédio de duas gigantes: Gradiente e Epcom (da Sharp); a primeira com o Expert XP800 e a segunda, o Hot Bit HB 8000.

Expert

Fabricante: Gradiente Informática (Tels.: (011) 881-9758 e (011) 881-9513).
Microprocessador: Z80A com clock de 3,58 Mhz.
Memória residente: ROM, 32 Kb; RAM, 80 Kb, sendo 16 Kb específicos para vídeo e 64 Kb disponíveis para o usuário.
Teclado: (destacável) alfanumérico semiprofissional, com 89 teclas com caracteres em português, gráficos, dez funções programáveis e bloco de comando do cursor independente.
Linguagem de programação: MSX BASIC.

Vídeo
Processador: TMS-9128NL.
Modo texto: 32 colunas x 24 linhas ou 40 colunas x 24 linhas.
Modo gráfico: 64 x 48 pontos ou 256 x 192 pontos, com 16 cores.

Som
Processador: AY-3-8910-A, com oito oitavas e três canais.

Saídas
Impressora: interface paralela, padrão Centronics.
Monitor de som: alto-falante interno com controle de volume externo.
Gravador: velocidade de transmissão 1200/2400 bauds, processo FSK.
Vídeo: duas saídas no sistema Pal-M (televisor e vídeo compostos e uma saída RGB).
Joystick: dois conectores frontais de nove pinos.
Slots: dois slots frontais para conexão de cartuchos, expansões e periféricos. O equipamento possui também um slots localizado na parte traseira, que funciona em conjunto com um dos slots dianteiros.
Tomada: possui duas tomadas suplementares para alimentação de periféricos controladas pela chave liga/desliga da UCP.
Alimentação: 120/220 V AC, 60 Hz.
Dimensões: UCP 420 x 110 x 280mm (LAP).
Teclado: 420 x 47 x 170mm (LAP).
Peso: UCP - 4,7Kg e teclado - 1,0Kg.

Hot Bit

Fabricante: EPCOM – Equipamentos Eletrônicos da Amazônia (tel.: (011) 211-7022 e (011) 211-9461).
Microprocessador: Z80A, com clock de 3,57 Mhz.
Memória residente: ROM, 32 Kb; RAM, 80 Kb, sendo 16 Kb específicos para vídeo e 64 Kb disponíveis para o usuário. Expansão possível até 512 Kb.
Teclado: (incluído no equipamento) alfanumérico semiprofissional de 73 teclas, com caracteres em português, seis funções pré-programadas, dez funções definíveis pelo usuário e bloco de comando de cursor independente.
Linguagem de programação: MSX BASIC.

Vídeo
Processador: TMS-9128NL.
Modo texto: 32 colunas x 24 linhas ou 40 colunas x 24 linhas.
Modo gráfico: 64 x 48 pontos ou 256 x 192 pontos, com 16 cores.

Som
Processador: AY-3-8910-A, com oito oitavas e três canais.

Saídas
Impressora: interface paralela, padrão Centronics.
Gravador: velocidade de transmissão 1200/2400 bauds, no formato FSK.
Vídeo: duas saídas no sistema PAL-M (televisor e vídeo compostos).
Joystick: dois conectores frontais de nove pinos.
Slots: dois slots (um na lateral do equipamento) para conexão de cartuchos, expansões e outros periféricos.
Tomada: possui uma tomada 120/220 V para alimentação de periféricos.
Alimentação: 120/220 V AC, 60 Hz.
Dimensões: 405 x 68 x 280mm (LAP).
Peso: 3,3Kg.

O MERCADO

A título de curiosidade é interessante notar que, no ano de 1984, a maior fatia do mercado japonês era dominada por quatro empresas, e cada uma produzia 10 mil MSX por mês!

A cada dia, o padrão MSX ganha adeptos em indústrias de todo o mundo. Com relação à sua aceitação no exterior (Europa e Japão), temos notícias que já foram vendidos mais de três milhões de micros!

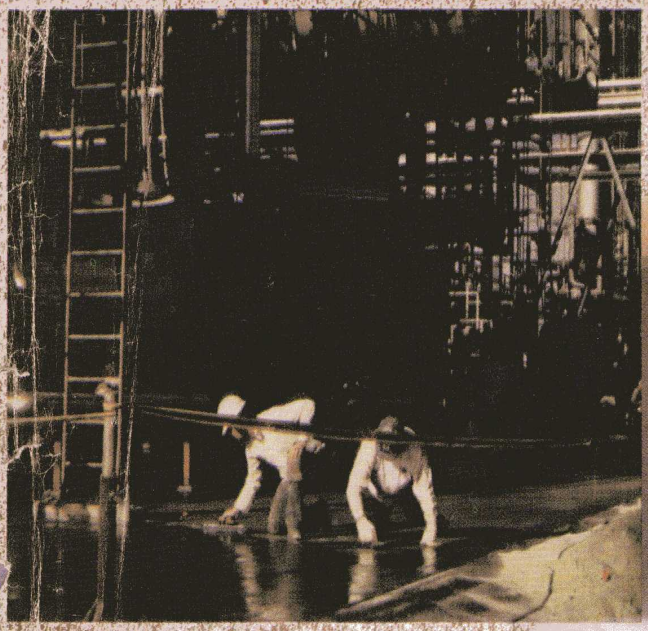
Tudo indica que num futuro próximo, os MSX nacionais ocuparão uma fatia significativa dos mercados doméstico e educacional, hoje ainda incipientes e pouco explorados.

Provavelmente serão desenvolvidos e lançados modelos mais sofisticados nos próximos anos, apresentando características como: melhor resolução gráfica e maior memória de vídeo; light pen incorporada ao micro; unidades de disco de três em três e 1/2" embutidas e 80 colunas em modo de texto.

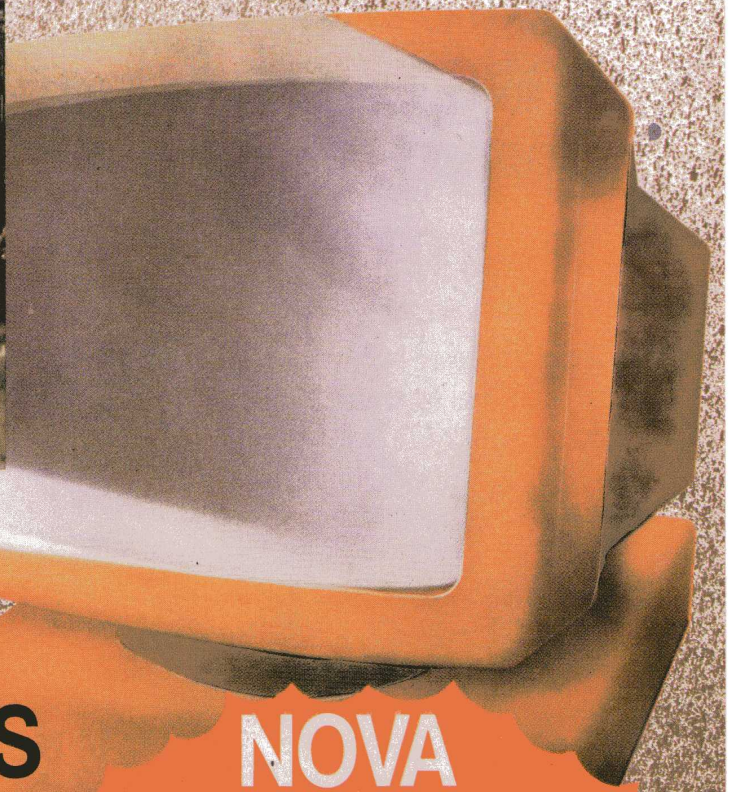
Cabe lembrar que todo e qualquer lançamento dentro da família MSX será compatível, tanto a nível de hardware como software, com os já existentes no mercado.

Como se vê, o assunto é longo e fascinante; continuaremos, pois, com ele nas próximas edições de **MS**, falando sobre o BASIC da Microsoft e dos aplicativos e jogos para este novo e revolucionário padrão. ■

Oscar Júlio Burd e Luiz Sérgio Y. Moreira, ambos formados em Física pela USP, são diretores do CEBI – Centro Educacional Brasileiro de Informática, empresa especializada na produção de software educacional e assessoria a colégios. O CEBI desenvolve ainda programas que exploram os recursos sonoros do MSX, como o programa TOQUE.



QUANTUM®



PARA SOLUÇÕES DEFINITIVAS

NOVA TECNOLOGIA EM RESINAS.

Trata-se de um sistema anticorrosivo e antiabrasivo com resina produzida sob avançada tecnologia nos Estados Unidos, Japão e agora no Brasil.

ONDE SE APLICA

- pisos de indústrias químicas, alimentícias, eletro-eletrônicas, automobilísticas, petroquímicas, armazéns, supermercados, hangares, oficinas mecânicas;
- sinalização de tráfego e de áreas industriais expostas a condições climáticas, ataques químicos e mecânicos;
- tanques de concreto ou aço;
- esgotos e efluentes industriais, municipais e residenciais;
- impermeabilização de paredes, pisos, tanques e torres de concreto com trincas em sua estrutura.

SUAS PRINCIPAIS VANTAGENS

- excepcional resistência a ataques químicos e abrasivos;
- cura rápida, entregando a área revestida em poucas horas ao tráfego de empilhadeiras;
- excelente ancoragem - coeficiente de expansão igual ao do concreto, tornando-se monolítica com a base, sem apresentar trincas ao longo da utilização;
- excepcionais resistências químicas (mais de 250 produtos químicos);
- excepcionais resistências mecânicas (impacto, compressão, flexão e abrasão);
- necessidade de piso anti-derrapante;
- fácil limpeza, não permitindo o desenvolvimento de bactérias patogênicas, vírus e fungos;
- temperatura de aplicação desde - 30°C até 60°C, não exigindo parada da área operacional.

COMO SE APLICA

A Resina Polymórfica Quantum pode ser aplicada na forma de pintura com pincéis, brochas, rolos, pistola dois componentes, ou na forma de argamassa quando a resina é aglomerada com areia sílica classificada.

QUANTUM

Produzida e Distribuída no Brasil pela
REVESTIMENTOS POLYPLEX INTERNACIONAL LTDA.
Av. São Luiz, 187 - 2º sobreloja - loja 25
01046 - São Paulo - SP - Fone: (011) 258-6609 - Telex: (011) 53728

para maiores informações, solicite nosso catálogo ou a visita dos nossos técnicos.

A Microdigital lança no Brasil o micro pessoal de maior sucesso no mundo.

A partir de agora a história dos micros pessoais vai ser contada em duas partes: antes e depois do TK 90X.

O TK 90X é, simplesmente, o único micro pessoal lançado no Brasil que merece a classificação de "software machine": um caso raro de micro que pela sua facilidade de uso, grandes recursos e preço acessível recebeu a atenção dos criadores de programas e periféricos em todo o mundo.

Para você ter uma idéia, existem mais de 2 mil programas, 70 livros, 30 periféricos e inúmeras revistas de usuários disponíveis para ele internacionalmente.

E aqui o TK 90X já sai com mais de 100 programas, enquanto outros estão em fase final de desenvolvimento para lhe dar mais opções para trabalhar, aprender ou se divertir que com qualquer outro micro.

O TK 90X tem duas versões de memória (de 16 ou 48 K), imagem de alta resolução gráfica com 8 cores, carregamento rápido de programas (controlável pelo próprio monitor), som pela TV, letras maiúsculas e minúsculas e ainda uma exclusividade: acentuação em português.

Faça o seu programa: peça já uma demonstração do novo TK 90X.

MICRODIGITAL

Chegou o micro cheio de programas.

TK 90X

Filiada à ABICOMP